



TUGAS AKHIR - TE 141599

**IMPLEMENTASI DAN ANALISA TEKNIK REDUKSI
PAPR OFDM MENGGUNAKAN METODE CLIPPING
DAN FILTERING PADA WARP**

FAHIMAH RAHMADIAN
NRP 2212 100 042

Dosen Pembimbing
Dr. Ir. Titiek Suryani, MT.
Dr. Ir. Suwadi, MT.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



FINAL PROJECT - TE 141599

**IMPLEMENTATION AND ANALYSIS OF PAPR
REDUCTION TECHNIQUE IN OFDM USING
CLIPPING AND FILTERING METHOD ON WARP**

**FAHIMAH RAHMADIAN
NRP 2212 100 042**

**Lecture Advisor
Dr. Ir. Titiek Suryani, MT.
Ir. Suwadi, MT.**

**ELECTRICAL ENGINEERING MAJOR
Industrial Technology Faculty
Sepuluh Nopember Institute of Technology**

Surabaya 2016

LEMBAR PENGESAHAN

IMPLEMENTASI DAN ANALISA TEKNIK REDUKSI PAPR OFDM MENGGUNAKAN METODE CLIPPING DAN FILTERING PADA WARP

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik

Pada

Bidang Studi Telekomunikasi Multimedia

Jurusan Teknik Elektro

Institut Teknologi Sepuluh Nopember

Menyetujui,

Dosen Pembimbing I



Dr. Ir. Titiek Suryani, MT.
NIP. 196411301989032001

Dosen Pembimbing II



Dr. Ir. Suwadi, MT.
NIP. 196808181993031002



IMPLEMENTASI DAN ANALISA TEKNIK REDUKSI PAPR OFDM MENGGUNAKAN METODE CLIPPING DAN FILTERING PADA WARP

Nama : Fahimah Rahmadian
Pembimbing : Dr. Ir. Titiek Suryani, MT.
Dr. Ir. Suwadi, MT.

ABSTRAK

Orthogonal Frequency Division Multiplexing (OFDM) sebagai salah satu teknik multicarrier memiliki kelemahan yaitu nilai perbandingan daya puncak dengan daya rata-rata sinyal atau *Peak to Average Power Ratio* (PAPR). Nilai PAPR yang tinggi akan menyebabkan sistem rentan terhadap distorsi nonlinear ketika melewati *Power Amplifier* (PA) yang dapat mengakibatkan nilai BER naik dan menyebabkan performansi OFDM akan menurun.

Teknik *clipping dan filtering* adalah teknik reduksi PAPR yang sederhana dan tidak memiliki tingkat kompleksitas yang tinggi. Pada tugas akhir ini akan diimplementasikan dan dianalisa teknik reduksi PAPR sinyal OFDM menggunakan metode *clipping* dan *filtering* pada WARP yang merupakan salah satu jenis *Software Defined Radio* (SDR). Parameter yang digunakan adalah SNR untuk simulasi dan *gain* pemancar yang akan digunakan untuk pengukuran pada WARP. Nilai yang akan diukur adalah BER dan PAPR, di mana BER yang didapatkan dari hasil *clipping filtering* nilainya lebih baik jika dibandingkan dengan BER pada sistem OFDM. Namun, BER yang dihasilkan dari proses *clipping* saja hasilnya lebih tinggi daripada BER sistem OFDM akibat pengaruh radiasi in band. Metode *clipping filtering* dapat menurunkan nilai PAPR sebesar 29.43% sampai 45.65%.

Kata kunci: OFDM, PAPR, *Clipping*, *Filtering*, WARP

IMPLEMENTATION AND ANALYSIS OF PAPR REDUCTION TECHNIQUE IN OFDM USING CLIPPING AND FILTERING METHOD ON WARP

Name : Fahimah Rahmadian
Supervisors : Dr. Ir. Titiek Suryani, MT
Dr. Ir. Suwadi, MT.

ABSTRACT

Orthogonal Frequency Division Multiplexing (OFDM) as one of the multicarrier technique has a main drawback which is the ratio between peak power and average power or Peak to Average Power Ratio (PAPR). A high value of PAPR can cause the system more vulnerable to nonlinear distortion while passing Power Amplifier (PA). A high value of PAPR can cause the BER value increase and the OFDM performance decrease.

Clipping and filtering technique is a simple PAPR reduction technique and has a low complexity level. In this thesis, PAPR reduction using clipping and filtering method will be implemented and analyzed in WARP which is one of type of Software Defined Radio (SDR). Parameters that are used are SNR for simulation and transmitter's power in WARP. The values to be measured are BER and PAPR. BER is obtained from the clipping filtering has a better value than BER on the OFDM system. However, the BER resulting from the clipping process is higher than BER OFDM system due to the effect of radiation in band. Clipping method of filtering can reduce PAPR value by 29.43% to 45.65%.

Keywords: OFDM, PAPR, Clipping, Filtering, WARP

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metodologi	2
1.6 Sistematika Penulisan	4
1.7 Manfaat	4
BAB 2 TEORI PENUNJANG	5
2.1 Orthogonal Frequency Division Multiplexing (OFDM)	5
2.2 M-ary Quadrature Amplitude Modulation	8
2.2.1 16-ary QAM	8
2.2.2 64-ary QAM	9
2.3 Peak to Average Power Ratio (PAPR)	9
2.4 Metode Clipping dan Filtering	11
2.5 Distorsi Non Linier Model Saleh	13
2.6 Wireless Open-Access Research Platform (WARP)	15
2.6.1 Arsitektur Platform	16
2.6.2 Design Arsitektur WARPLab	17
2.6.3 Gambaran Framework WARPLab	18
2.6.4 Spesifikasi WARPLab	19
BAB 3 DESAIN DAN SIMULASI SISTEM	21
3.1 Alur Pengerjaan Tugas Akhir	21
3.2 Desain Sistem OFDM	22
3.2.1 Sistem OFDM pada WARP	22
3.2.2 Estimasi Kanal	28
3.2.3 Estimasi Fasa Error	29
3.3 Metode Reduksi PAPR (Clipping dan Filtering)	31
3.4 Perhitungan PAPR	32
3.5 Perhitungan BER	33
3.6 Implementasi Sistem pada WARP	33

3.6.1 Matlab R2014a.....	33
3.6.2 Setting TCP/IP	33
3.6.3 Kelengkapan Perangkat Keras	34
3.6.4 Langkah-langkah Konfigurasi Awal Program pada WARP ..	36
3.7 Skenario Pengukuran	36
BAB 4 PENGUKURAN DAN ANALISA DATA.....	39
4.1 Sinyal yang Dikirim dan Diterima	39
4.2 Korelasi LTS.....	41
4.3 Estimasi	42
4.4 Scatterplot Sinyal	43
4.5 Pengaruh Clipping Ratio (CR) Terhadap Nilai BER dan PAPR	44
4.5.1 Analisa Perbandingan BER.....	45
4.5.2 Analisa Perbandingan PAPR	47
4.6 Analisa Bit Error Rate (BER)	50
4.6.1 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping	50
4.6.2 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping Filtering.....	53
4.7 Analisa Peak to Average Power Ratio (PAPR)	59
BAB 5 PENUTUP	61
5.1 Kesimpulan	61
5.2 Saran	61
DAFTAR PUSTAKA.....	63
LAMPIRAN	65
RIWAYAT HIDUP.....	98

DAFTAR TABEL

Tabel 4.1	Parameter Simulasi Reduksi PAPR pada OFDM.....	39
Tabel 4.2	Nilai BER pada Simulasi dengan Modulasi 16QAM.....	45
Tabel 4.3	Nilai BER pada Simulasi dengan Modulasi 64QAM.....	46
Tabel 4.4	Reduksi PAPR untuk Modulasi 16QAM	47
Tabel 4.5	Reduksi PAPR untuk Modulasi 16QAM	48
Tabel 4.6	Nilai BER Sistem OFDM dan Teknik Clipping pada Simulasi.....	51
Tabel 4.7	Nilai BER Sistem OFDM dan Teknik Clipping Pengukuran pada WARP.....	53
Tabel 4.8	Nilai BER Sistem OFDM dan Teknik Clipping Filtering	55
Tabel 4.9	Nilai BER Sistem OFDM dan Teknik Clipping Filtering pada Pengukuran.....	57
Tabel 4.10	Nilai PAPR Sistem OFDM dan Teknik Clipping Filtering.....	60

DAFTAR GAMBAR

Gambar 2.1	(a) Modulasi FDM (b) Modulasi OFDM.....	5
Gambar 2.2	Blok Diagram <i>Transmitter</i> dan <i>Receiver</i> dalam OFDM.....	7
Gambar 2.3	Diagram Konstelasi 16-QAM	8
Gambar 2.4	Diagram Konstelasi 64-QAM	9
Gambar 2.5	Representasi <i>Peak Power</i> dengan <i>Average Power</i>	10
Gambar 2.6	Diagram Blok <i>Clipping</i> dan <i>Filtering</i>	12
Gambar 2.7	Subsistem Nonlinearitas	13
Gambar 2.8	Konversi AM/AM dan AM/PM pada Model Saleh.....	14
Gambar 2.9	Diagram Konstelasi 16-QAM (a) Tanpa Efek Nonlinearitas (b) Dengan Efek Nonlinearitas.....	14
Gambar 2.10	Komponen – komponen Platform	16
Gambar 2.11	Design Arsitektur WARPLab.....	17
Gambar 3.1	<i>Flowchart</i> Pengerjaan Tugas Akhir.....	21
Gambar 3.2	Diagram Blok Sistem OFDM pada WARP	22
Gambar 3.3	Konversi Serial ke Paralel	23
Gambar 3.4	Penempatan Data dan Pilot pada Blok IFFT	24
Gambar 3.5	Susunan <i>Preamble</i>	25
Gambar 3.6	Blok Diagram Interpolasi	25
Gambar 3.7	Blok Diagram Desimasi	26
Gambar 3.8	Konversi Paralel ke Serial	28
Gambar 3.9	Diagram Konstelasi Estimasi Kanal	30
Gambar 3.10	Perbandingan Fasa Sinyal dengan Koreksi Fasa	31
Gambar 3.11	Blok Diagram <i>Clipping Filtering</i> pada Sistem OFDM ...	31
Gambar 3.12	Script Matlab Perhitungan PAPR.....	32
Gambar 3.13	Pengaturan <i>IP Address</i> untuk WARP	34
Gambar 3.14	Kelengkapan Perangkat WARP	35
Gambar 3.15	Konfigurasi Perangkat Pengukuran	35
Gambar 3.16	Sketsa pengukuran <i>indoor</i>	37
Gambar 4.1	Sinyal yang <i>Dicliping</i> dengan $CR=0.85$	40
Gambar 4.2	Hasil Korelasi LTS.....	41
Gambar 4.3	Estimasi kanal (a) <i>Inphase & Quadrature</i> (b) <i>Magnitude</i>	42
Gambar 4.4	Estimasi Fasa <i>Error</i>	43
Gambar 4.5	Diagram Konstelasi 16QAM pada (a) Gain 50 (b) Gain 60	44
Gambar 4.6	Diagram Konstelasi 64QAM pada (a) Gain 50 (b) Gain 60	44

Gambar 4.7	Kurva Perbandingan BER dengan Peubah CR pada Modulasi 16QAM	45
Gambar 4.8	Kurva Perbandingan BER dengan Peubah CR pada Modulasi 64QAM	46
Gambar 4.9	Grafik Penurunan PAPR Akibat Pengaruh CR pada Modulasi 16QAM	48
Gambar 4.10	Grafik Penurunan PAPR Akibat Pengaruh CR pada Modulasi 64QAM	49
Gambar 4.11	Perbandingan BER Simulasi Sistem OFDM dan Metode <i>Clipping</i> pada (a) Modulasi 16QAM (b) Modulasi 64QAM	52
Gambar 4.12	Perbandingan BER Pengukuran Sistem OFDM dan Metode <i>Clipping</i> pada (a) Modulasi 16QAM (b) Modulasi 64QAM	54
Gambar 4.13	Perbandingan BER Simulasi Sistem OFDM dan Metode <i>Clipping Filtering</i> pada (a) Modulasi 16QAM (b) Modulasi 64QAM	56
Gambar 4.14	Perbandingan BER Pengukuran Sistem OFDM dan Metode <i>Clipping Filtering</i> pada (a) Modulasi 16QAM (b) Modulasi 64QAM	58
Gambar 4.15	Perbandingan Nilai PAPR Sistem OFDM dan Metode <i>Clipping Filtering</i> pada Modulasi 16QAM dan 64QAM	60

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi nirkabel atau teknologi elektronik yang beroperasi tanpa kabel dapat dimanfaatkan untuk komunikasi dan pengontrolan. Keberadaan berbagai sistem radio saat ini dituntut untuk dapat melakukan komunikasi dengan kecepatan data yang sangat tinggi. Hal ini yang membuat kebutuhan penggunaan spektrum dengan efisiensi spektral tinggi. Teknik *multicarrier* menjadi kandidat potensial untuk sistem nirkabel pita lebar karena keunggulannya dalam ketahanan terhadap distorsi kanal dan interferensi *narrowband*, efisiensi spektral yang tinggi, fleksibilitas yang tinggi, dan mendukung skema modulasi adaptif.

Teknik *multicarrier* yang kebanyakan digunakan adalah OFDM (*Orthogonal Frequency Division Multiplexing*). Pada OFDM, sinyal masukan akan dibagi ke dalam *subchannel-subchannel*, kemudian dengan menggunakan IFFT, sinyal output OFDM akan menjadi *orthogonal* satu sama lain. Sehingga bila ada distorsi pada jalur komunikasi yang menyebabkan ISI (*Inter Symbol Interference*) maka pada stasiun penerima, setiap *subchannel* masih dapat dipisahkan dengan menggunakan FFT.

Namun teknologi OFDM ini memiliki kelemahan yaitu tingginya nilai *Peak to Average Power Ratio* (PAPR), yaitu perbandingan antara daya puncak sinyal (*peak power*) dengan daya rata-rata sinyal (*average power*). Nilai PAPR yang tinggi akan menyebabkan distorsi nonlinear, penurunan *Signal-to-Quantization Noise Ratio* (SQNR) pada ADC dan DAC, dan menurunkan efisiensi *power amplifier* pada *transmitter* sehingga menyebabkan menurunnya performansi sistem OFDM. Untuk itu, diperlukan sebuah metode yang dapat mereduksi nilai PAPR sehingga kinerja sistem OFDM optimal.

Terdapat banyak metode yang dapat digunakan untuk mereduksi PAPR pada sinyal OFDM di antaranya *clipping* dan *filtering*, *tone injection*, *coding*, *selected mapping* (SLM), dan *Partial Transmit Sequence* (PTS). Pemilihan metode *clipping* dan *filtering* menurut penelitian yang telah dilakukan mampu menurunkan nilai reduksi PAPR dengan cara yang sederhana.

Sehingga pada tugas akhir ini akan dilakukan implementasi dengan menggunakan metode *clipping* dan *filtering* yang akan disimulasikan menggunakan *software* matlab dan diterapkan pada sistem OFDM berbasis *Wireless Open-Access Research Platform* (WARP) dan dilakukan pengukuran pada lingkungan indoor. Platform WARP merupakan salah satu jenis teknologi Software Defined Radio (SDR).

1.2 Perumusan Masalah

Masalah yang akan dibahas dalam Tugas Akhir ini adalah :

1. Bagaimana cara kerja *clipping* dan *filtering* dalam mereduksi nilai PAPR dalam OFDM?
2. Bagaimana pengimplementasian OFDM pada WARP?
3. Seberapa besar nilai *clipping ratio* yang diperbolehkan?
4. Seberapa besar perbaikan BER yang merupakan ketahanan sistem terhadap distorsi *nonlinear*?

1.3 Batasan Masalah

Permasalahan dibatasi sebagai berikut :

1. Teknik modulasi yang digunakan adalah 16-QAM dan 64-QAM.
2. Gain pemancar untuk mengamati distorsi non linier pada pengukuran adalah 50 dB sampai 60 dB, dengan selisih 2 dB.
3. Nilai SNR peubah pada simulasi adalah 0 dB sampai 60 dB, dengan selisih 10 dB.
4. Daerah pengukuran dibatasi dalam lingkungan *indoor*, yaitu di Laboratorium Komunikasi Multimedia B304

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah menganalisa teknik reduksi PAPR pada OFDM menggunakan teknik *clipping* dan *filtering* yang disimulasikan dengan *software* matlab, serta mengimplementasikannya pada WARP.

1.5 Metodologi

Dalam proses pengerjaan penelitian Tugas Akhir ini dapat dilakukan dengan mengelompokkan dalam beberapa metodologi, yaitu :

1. Studi Literatur

Pencarian dan pengumpulan literatur yang berkaitan dengan teknik reduksi PAPR pada OFDM menggunakan teknik *clipping* dan *filtering*, baik berupa jurnal artikel, buku referensi, internet, dan dari sumber yang lain.

2. Perancangan Teknik Reduksi PAPR

Pada tahap ini, dilakukan perancangan teknik reduksi PAPR menggunakan teknik *clipping* dan *filtering*. Perancangan tersebut berupa perancangan OFDM tanpa reduksi PAPR dan dengan reduksi PAPR. Setelah perancangan selesai, kemudian dilanjutkan dengan implementasi pada WARP.

3. Implementasi Teknik Reduksi PAPR

Setelah simulasi teknik reduksi PAPR menggunakan software matlab selesai, script tersebut akan disinkronisasikan dan diimplementasikan pada WARP, yang selanjutnya diikuti dengan proses pengukuran.

4. Pengukuran OFDM Dengan dan Tanpa Reduksi PAPR

Pada tahap ini, dilakukan pengukuran OFDM tanpa reduksi PAPR dan dengan reduksi PAPR menggunakan teknik *clipping* dan *filtering*.

5. Pengukuran Dengan Distorsi Non Linier

Pada tahap ini akan diberikan gangguan non linier ke dalam dua sistem yang telah dibuat sebelumnya. Gangguan non linier pada simulasi matlab menggunakan distorsi non linier model Saleh. Pada pengukuran, distorsi non linier berasal dari komponen yang ada di WARP.

6. Analisis Hasil

Hasil dari pengukuran akan dianalisis sejauh mana ketahanan sistem terhadap distorsi non linier menggunakan metode *clipping* dan *filtering*. Yang menjadi fokus parameter penelitian ini selain PAPR adalah perbaikan BER, *scatter plot*, dan spektrum. Kemudian dibandingkan dengan yang tidak menggunakan metode *clipping* dan *filtering*, untuk didapatkan kesimpulannya.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini akan diuraikan latar belakang, perumusan masalah, batasan masalah, tujuan, metodologi dan manfaat.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dibahas landasan teori mengenai konsep OFDM, modulasi QAM M-ary, *Peak to Average Power Ratio* (PAPR), teknik reduksi PAPR menggunakan *clipping* dan *filtering*, serta WARP.

BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini akan dijelaskan tentang perancangan dan implementasi sistem teknik reduksi PAPR dengan *clipping* dan *filtering*. Metode *clipping* dan *filtering* disusun dengan *software* matlab dan diimplementasikan pada WARP.

BAB IV PENGUJIAN DAN ANALISIS

Pada bab ini berisi hasil dari implementasi yang sudah di buat pada bab III. Dari pengujian ini kemudian dianalisis dan ditarik kesimpulan sementara mengenai parameter-parameter yang telah diuji.

BAB V PENUTUP

Pada bab ini berisi kesimpulan dari keseluruhan materi dan dari hasil analisis data pada bab IV. Selain itu pada bab ini dibahas mengenai saran yang bisa dilakukan untuk pengembangan penelitian selanjutnya.

1.7 Manfaat

Hasil yang diperoleh dari tugas akhir ini diharapkan dapat memberi manfaat, yakni :

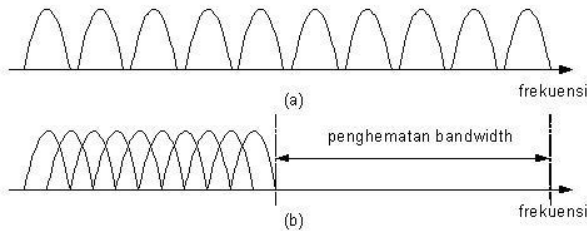
1. Memahami proses dan cara kerja teknik reduksi PAPR pada OFDM menggunakan teknik *clipping* dan *filtering* dengan menggunakan kanal real pada WARP
2. Memberikan kontribusi tentang proses desain dan implementasi teknik reduksi PAPR menggunakan teknik *clipping* dan *filtering* pada WARP

BAB 2

TEORI PENUNJANG

2.1 Orthogonal Frequency Division Multiplexing (OFDM)

Orthogonal Frequency Division Multiplexing (OFDM) merupakan teknik transmisi yang digunakan sistem komunikasi digital yang menggunakan beberapa frekuensi sehingga sinyal yang dikirimkan akan saling orthogonal satu dengan yang lain. OFDM memecah data serial berkecepatan tinggi menjadi data paralel dengan kecepatan yang lebih rendah, sehingga data-data terbagi dalam beberapa subkanal. OFDM menggunakan prinsip modulasi *subcarrier*. Gambar 2.1 (a) menampilkan spektrum frekuensi pada modulasi FDM dan gambar 2.1 (b) menampilkan spektrum frekuensi pada modulasi OFDM.



Gambar 0.1 (a) Modulasi FDM (b) Modulasi OFDM

Orthogonalitas harus dipenuhi supaya didapatkan efisiensi spektral yang lebih baik dan kapasitas kanal transmisi yang maksimum. Tujuan mempertahankan orthogonalitas adalah untuk mencegah *overlapping* frekuensi tengah dari subcarrier supaya spektrum *subcarrier* dapat *overlap* agar mendapatkan efisiensi spektral yang lebih tinggi. Untuk memeriksa apakah dua subcarrier dalam OFDM ortogonal, dapat diketahui dari persamaan berikut:

$$\begin{aligned} \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi f_k t} e^{j2\pi f_l t} dt &= \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi \frac{k}{T_{sym}} t} e^{j2\pi \frac{l}{T_{sym}} t} dt \\ &= \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi \frac{(k-l)}{T_{sym}} t} dt \end{aligned} \quad (2.1)$$

Jika hasil perkalian dot dari sinyal tidak sama dengan nol, maka dua sinyal tidak memenuhi syarat ortogonalitas, sebaliknya jika tidak maka dua sinyal dikatakan ortogonal.

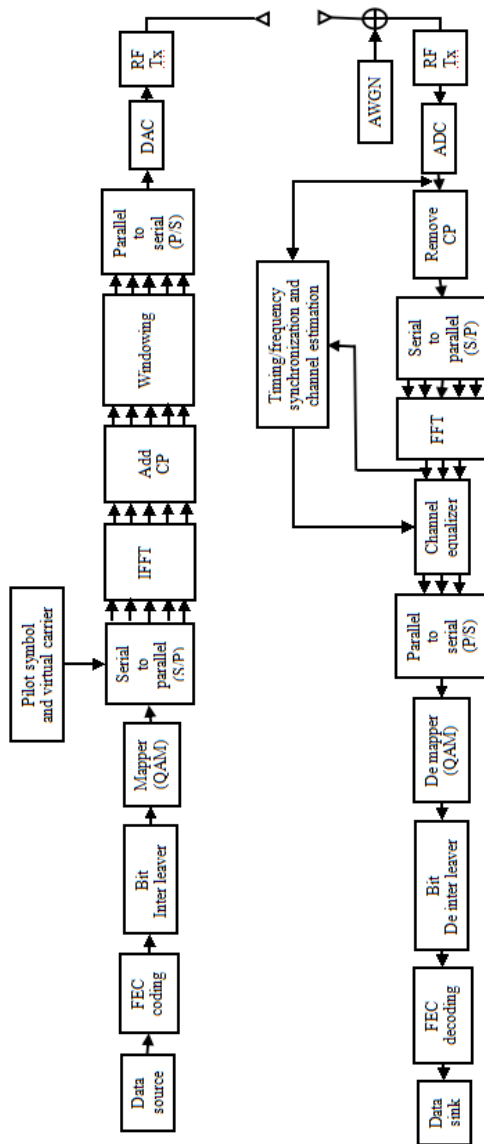
Untuk mencegah *Inter Symbol Interference* (ISI), *guard interval* digunakan dan disisipkan di antara simbol OFDM untuk menghilangkan ISI yang dapat diakibatkan oleh delay transmisi. *Guard interval* yang digunakan disini yaitu *cyclic prefix* (CP).

Keuntungan OFDM bila dibandingkan dengan *single carrier* dan *multicarrier* konvensional FDM, di antaranya adalah OFDM lebih efisien dalam pemakaian frekuensi di mana antarfrekuensi yang bersebelahan diperbolehkan karena masing-masing telah saling orthogonal sedangkan pada sistem *multicarrier* konvensional untuk mencegah interferensi antarfrekuensi yang bersebelahan perlu disisipkan guard band. OFDM tahan dalam menghadapi frequency selective fading dan tidak sensitif terhadap sinyal delay karena rendahnya kecepatan transmisi di setiap subcarrier yang berarti periode simbolnya menjadi lebih panjang sehingga sensitivitas terhadap delay spread atau penyebaran sinyal-sinyal yang delay relatif berkurang.

Sedangkan beberapa kelemahan dari teknologi OFDM selain besarnya nilai PAPR, di antaranya adalah frequency offset yang disebabkan oleh jitter pada *carrier wave* dan juga terhadap efek doppler yang disebabkan oleh pergerakan transmitter maupun receiver, OFDM mudah terkena distorsi non linier yang terjadi pada *amplifier* dari daya transmisi.[3]

Gambar 2.2 menjelaskan tentang *transmitter* dan *receiver* dalam OFDM. Data akan dibangkitkan melalui proses *mapping* dan akan berupa data digital 1 dan 0. OFDM memecah data serial berkecepatan tinggi menjadi data paralel dengan kecepatan yang lebih rendah. Lalu, dalam proses IFFT, simbol pilot dan *virtual carrier* akan disisipkan untuk mengurangi terjadinya ISI. Penambahan *cyclic prefix* juga terjadi setelah proses IFFT. Data yang tadinya berbentuk paralel akan diubah ke dalam bentuk serial dan selanjutnya masuk ke dalam DAC karena agar siap ditransmisikan, sinyal harus berada dalam bentuk analog.

Pada penerima, data akan masuk ke dalam ADC sehingga data berbentuk digital. Lalu, *cyclic prefix* yang telah ditambahkan sebelumnya akan dihilangkan dan dilanjutkan dengan proses serial ke paralel. Setelah itu, sinyal akan melalui proses FFT dan diubah lagi ke bentuk serial untuk selanjutnya dilakukan proses *demapper* sehingga didapatkan sinyal yang diinginkan.



Gambar 0.2 Blok Diagram Transmitter dan Receiver dalam OFDM

2.2 M-ary Quadrature Amplitude Modulation

M-ary diambil dari kata *binary*. M merepresentasikan digit yang mengartikan jumlah kondisi, level, atau kemungkinan kombinasi jumlah variabel biner tertentu. Jumlah bit yang diperlukan untuk menghasilkan jumlah kondisi diekspresikan dalam persamaan berikut:

$$N = \log_2 M \quad (2.2)$$

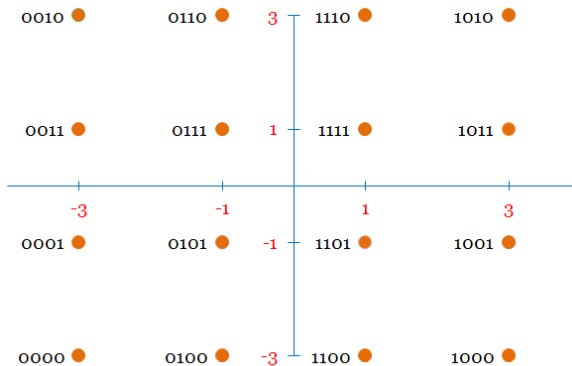
dengan:

N : jumlah bit

M : jumlah level, kondisi, atau keduanya

2.2.1 16-ary QAM

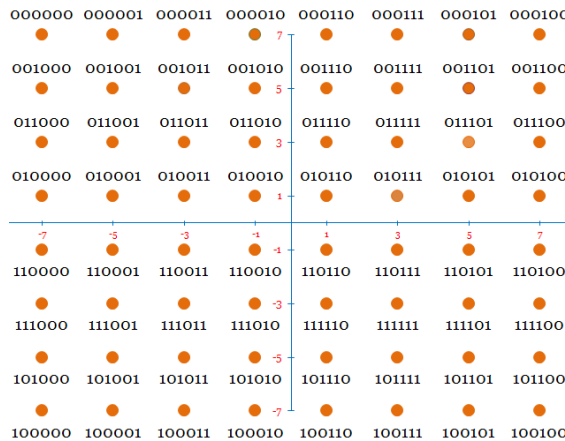
16-QAM adalah teknik modulasi di mana 16 adalah simbol yang mungkin dengan amplitudo dan fasa yang berbeda. Data masukan dibagi menjadi 4 bit ($2^4 = 16$), atau disebut quadbit. Data masukan biner dibagi menjadi 4 kanal yaitu I, I', Q, Q'. Laju bit pada masing-masing kanal sebesar 1/4 dari laju bit masukan ($f_b/4$). Empat bit masukan secara serial sampai pemecahan bit, dan dikeluarkan secara serentak dan paralel pada kanal I, I', Q, Q'. Bit pada kanal I dan Q menyatakan polaritas dari konverter (logika 1=+V, logika 0=-V), bit pada kanal I' dan Q' menyatakan besar keluaran (logika 1=0.821 V dan logika 0=0.22 V).



Gambar 0.3 Diagram Konstelasi 16-QAM

2.2.2 64-ary QAM

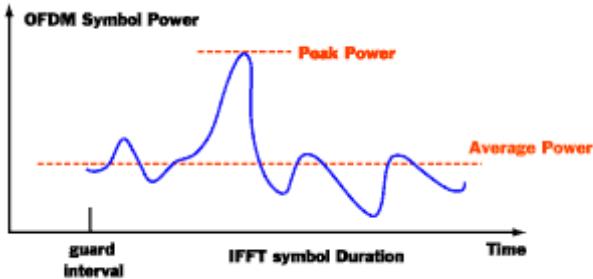
64-QAM adalah teknik modulasi dengan $M=64$ di mana ada 64 keluaran yang mungkin dengan amplitudo dan fasa yang berbeda. Data masukan biner dibagi menjadi 6 bit ($2^6=64$) atau disebut heksabit. Data maksimum biner dibagi menjadi 6 kanal yaitu: Q, Q', Q'', I, I', I''. Bit pada kanal I dan Q menyatakan polaritas dari konverter (logika 1= $+V$, logika 0= $-V$), sedangkan bit pada kanal Q', Q'' dan I', I'' menyatakan besar keluaran (00=0,821 V; 01=0,22 V; 10=1,307 V, dan 11=0,541 V).



Gambar 0.4 Diagram Konstelasi 64-QAM

2.3 Peak to Average Power Ratio (PAPR)

Kendala yang dihadapi dalam sistem OFDM adalah nilai PAPR yang tinggi. PAPR adalah perbandingan antara daya puncak sinyal (*Peak Power*) dengan daya rata-rata sinyal (*Average Power*). PAPR dapat terjadi sebagai hasil superposisi dari dua atau lebih sinyal informasi sehingga menghasilkan nilai *peak* sinyal yang sangat besar. Hal ini biasanya disebabkan oleh modulasi masing-masing *subcarrier* yang dilakukan dengan frekuensi yang berbeda sehingga menyebabkan beberapa *subcarrier* mempunyai fasa koheren yang pada akhirnya akan muncul amplitudo dengan level yang jauh lebih besar dari daya sinyalnya, yang selanjutnya disebut dengan PAPR.



Gambar 0.5 Representasi Peak Power dengan Average Power

PAPR dari sinyal analog yang ditransmisikan dapat didefinisikan sebagai:

$$PAPR = \frac{\max |s(t)|^2}{E[|s(t)|^2]} \quad (2.3)$$

Dengan:

$\max |s(t)|^2$: puncak sinyal OFDM
 $E[|s(t)|^2]$: rata-rata sinyal OFDM

Di mana batasan waktu yang diamati ada pada suatu interval tertentu. Umumnya, nilai PAPR dihitung untuk satu simbol OFDM. Dengan cara yang sama, PAPR sinyal waktu diskrit bisa didefinisikan sebagai:

$$PAPR = \frac{\max |s(n)|^2}{E[|s(n)|^2]}, 1 \leq n \leq N \quad (2.4)$$

Di mana $E[|s(n)|^2]$ menyatakan daya rata-rata. PAPR biasanya diekspresikan dalam dB.

Banyak teknik reduksi PAPR dari sinyal OFDM yang telah dikembangkan saat ini. Teknik-teknik tersebut melakukan reduksi PAPR dengan pengorbanan seperti distorsi sinyal original, peningkatan daya, penurunan efisiensi bandwidth, peningkatan BER, penurunan *data rate*, peningkatan kompleksitas perhitungan, kebutuhan *side information* dan lain-lain.

PAPR yang tinggi akan mengakibatkan kerugian seperti distorsi pada perangkat *analog-to-digital* dan *digital-to-analog konverter* dan menurunkan efisiensi *amplifier* daya (*power amplifier*).

Beberapa skema telah diajukan untuk mengurangi PAPR. Secara garis besar skema-skema tersebut dapat dikelompokkan menjadi tiga kategori :

- a. Metode distorsi atau pencacatan sinyal
Metode ini meliputi *clipping* dan *filtering*, *active constellation extension*, *peak windowing*, *peak cancellation*, dan *peak power suppression*.
- b. Metode *distortionless* atau probabilistik
Metode ini meliputi *multiple signal representation* (MSR), *selective mapping* (SLM), dan *partial transmit sequence* (PTS).
- c. Metode koding atau pengacakan sinyal (*signal scrambling*)
Metode ini meliputi *Golay complementary sequences*, *Shapiro-Rudin sequences*, *M-sequences*, dan *Barker codes*.

2.4 Metode Clipping dan Filtering

Nilai PAPR yang besar akan mengakibatkan sebaran spektrum yang signifikan (*out-of-band*), *intersymbol interference* (ISI) dan distorsi *in-band* ketika dilewatkan melalui divais nonlinear. Hal tersebut dapat diatasi dengan menggunakan *amplifier* linear atau dengan *mem-backoff* titik operasi suatu *amplifier* nonlinear yang mengakibatkan menurunnya efisiensi daya.

Cara yang paling mudah untuk mengatasi PAPR yang tinggi adalah dengan memotong (*clipping*) sinyal masukan sebelum ke *amplifier*. Karena probabilitas terjadinya sinyal dengan *peak* yang tinggi sangat kecil, maka *clipping* merupakan suatu teknik yang efektif untuk menurunkan PAPR. Namun, *clipping* merupakan proses yang nonlinear dan akan mengakibatkan distorsi *in-band* yang cukup signifikan yang akibatnya akan menurunkan BER.

Daya puncak dari sinyal OFDM akan dipotong sesuai *threshold* menurut persamaan di bawah:

$$\begin{aligned}
 x(t) &= s(t), \text{ untuk } |s(t)| \leq A \\
 x(t) &= Ae^{j\varphi t}, \text{ untuk } |s(t)| > A \\
 \varphi &= \arg(s(t))
 \end{aligned}
 \tag{2.5}$$

Di mana:

$x(t)$ = simbol OFDM yang telah dipotong

$s(t)$ = simbol OFDM

A = nilai *threshold*

Langkah selanjutnya adalah menentukan nilai maksimal dari proses *clipping* itu sendiri. Nilai dari *clipping ratio* (CR) ditentukan terlebih dahulu dengan persamaan di bawah:

$$CR = \frac{A_{\max}}{\sigma} \quad (2.6)$$

Di mana CR adalah clipping ratio, A_{\max} adalah level clipping, dan σ adalah nilai RMS dari sinyal OFDM.

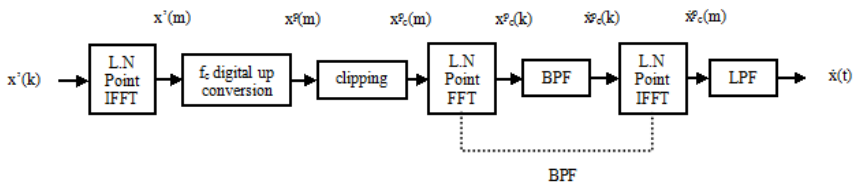
Setelah itu, gunakan persamaan di atas untuk menentukan nilai maksimal atau threshold dengan persamaan berikut ini:

$$\sigma = \sqrt{N_c}$$

$$A_{\max} = CR \sqrt{N_c} \quad (2.7)$$

N_c = jumlah *subcarrier*

Berikut adalah diagram blok dari *clipping* dan *filtering*:



Gambar 0.6 Diagram Blok Clipping dan Filtering

Dengan:

L : faktor *oversampling*

N : jumlah *subcarrier*

$x'(m)$: sinyal waktu diskrit

$x^p(m)$: sinyal passband (hasil modulasi sinyal diskrit dengan frekuensi carrier f_c)

$x^p_c(m)$: sinyal $x^p(m)$ yang telah *diclipping*

(((2.8)))(((9999999999

$$x^P(m) \quad \left| x^P(m) \right| < A$$

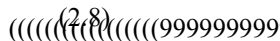
(((2.8)))(((9999999999

(((2.8)))(((9999999999

(((2.8)))(((9999999999

(((2.8)))(((9999999999

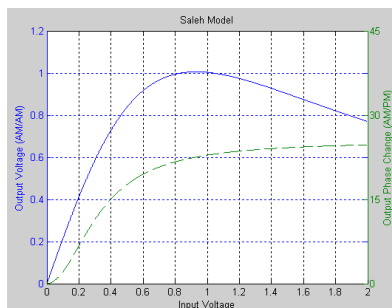
(((2.8)))(((9999999999



(((2.8)))(((9999999999

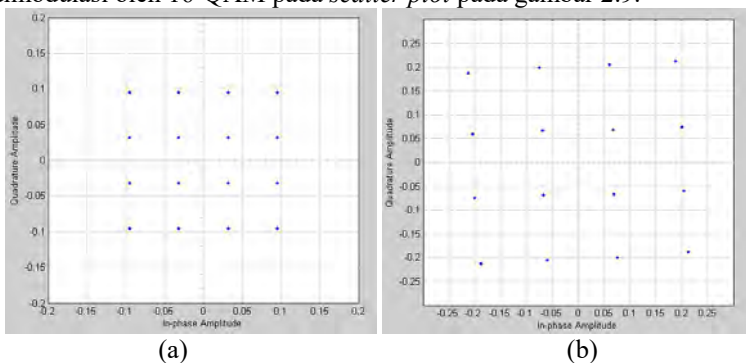
Masing-masing subsistem memberikan nonlinearitas ke sinyal input sebagai berikut:

- Sinyal dikalikan dengan faktor *gain*
- Sinyal kompleks dibagi ke dalam komponen magnitudo dan fase.
- AM/AM dikonversi ke magnitudo sinyal, sesuai metode yang dipilih, untuk menghasilkan magnitudo dari sinyal output
- AM/PM dikonversi ke fase sinyal, sesuai metode yang dipilih, untuk menghasilkan sudut dari sinyal output
- Magnitudo yang baru dan komponen sudut digabungkan ke sinyal kompleks dan hasilnya dikalikan dengan faktor *gain*, yang diatur oleh parameter gain linear



Gambar 0.8 Konversi AM/AM dan AM/PM pada Model Saleh

Efek dari nonlinearitas *memoryless* dapat dilihat pada sinyal yang dimodulasi oleh 16-QAM pada *scatter plot* pada gambar 2.9.



Gambar 0.9 Diagram Konstelasi 16-QAM (a) Tanpa Efek Nonlinearitas
(b) Dengan Efek Nonlinearitas

Parameter skala input (dB) mengindikasikan sinyal input sebelum nonlinearitas digunakan. Blok nonlinearitas mengalikan sinyal input dengan nilai parameter, diubah dari desibel ke satuan linear. Jika parameter diatur menjadi invers dari amplitudo sinyal input, sinyal yang dihitung memiliki amplitudo ternormalisasi sebesar 1.

Parameter AM/AM, α dan β , digunakan untuk menghitung gain sinyal input menggunakan persamaan berikut:

$$F_{AM/AM}(u) = \frac{\alpha * u}{1 + \beta * u^2} \quad (2.9)$$

di mana u adalah magnitudo dari sinyal yang dihitung.

Parameter AM/PM, α dan β , digunakan untuk menghitung perubahan fasa untuk sinyal input menggunakan persamaan di bawah ini:

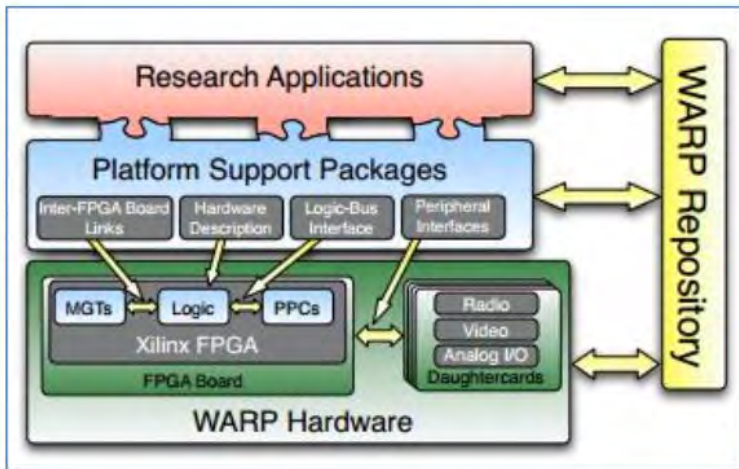
$$F_{AM/PM}(u) = \frac{\alpha * u^2}{1 + \beta * u^2} \quad (2.10)$$

Perlu diingat bahwa parameter AM/AM dan AM/PM, walaupun sama-sama dinamai α dan β , adalah berbeda. Parameter perhitungan output (dB) sama-sama mengindikasikan sinyal output.

2.6 Wireless Open-Access Research Platform (WARP)

Wireless Open-Access Research Platform (WARP) merupakan sebuah desain platform open-access tingkat tinggi yang sangat mampu, scalable, dan extensible untuk penelitian sistem komunikasi nirkabel. Platform WARP dirancang untuk memenuhi penelitian sistem komunikasi nirkabel dengan kinerja yang tinggi. Platform ini terdiri dari hardware dan implementasi FPGA yang keduanya merupakan blok kunci komunikasi. Dalam melakukan penelitian di bidang komunikasi nirkabel dengan pengembangan platform WARP ini, dapat secara bebas mendapatkan spesifikasi hardware dan implementasi algoritma karena telah tersedia secara bebas (Open Access).

2.6.1 Arsitektur Platform



Gambar 0.10 Komponen – komponen Platform

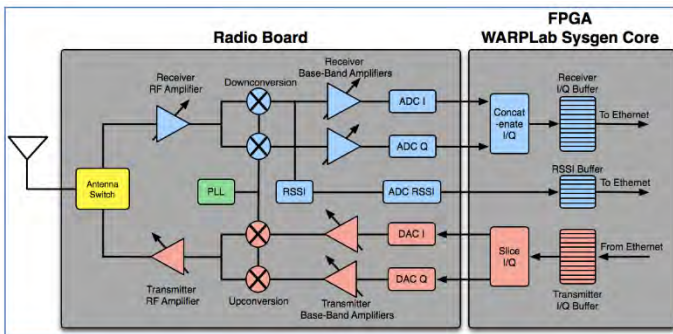
Platform ini didesain untuk memenuhi riset sistem komunikasi nirkabel dengan kinerja yang tinggi. Arsitektur platform ini terdiri dari empat komponen. Untuk lebih jelasnya bisa dilihat pada Gambar 2.11.

- *Custom Hardware* : Perangkat keras disesuaikan untuk memenuhi system komunikasi dengan kinerja yang tinggi. Perangkat keras ini menyediakan sumber daya untuk DSP berkecepatan tinggi yang bisa digunakan untuk implementasi algoritma yang intensif, terukur interkoneksi untuk mengalokasikan lebih banyak daya pemrosesan sehingga diperlukan prosesor khusus dan ekspansi interface yang lainnya.
- *Platform Support Packages* : alat desain dan modul antarmuka tingkat rendah dirancang untuk memungkinkan penggunaan hardware oleh para peneliti diseluruh lapisan desain jaringan nirkabel
- *Open-Access Repository* : pusat penyimpanan untuk semua *source code* model dan file desain hardware. *Content* secara lengkap akan tersedia di bawah lisensi BSD seperti open-source
- *Research Applications* : Implementasi dari algoritma baru, diaktifkan oleh hardware khusus dan paket mendukung platform.

Sistem kendali menggabungkan algoritma baru dapat dengan cepat dibangun menggunakan bangunan standar modul blok yang disediakan dalam repository.

2.6.2 Design Arsitektur WARPLab

Pada Gambar 2.11 menunjukkan jalur pemancar dan penerima dimana blok pemancar ditandai dengan warna merah muda dan blok penerima ditandai dengan warna biru.



Gambar 0.11 Design Arsitektur WARPLab

Blok pada jalur pemancar :

- Tx I/Q Buffer: sampel *In-Phase* yang akan dikirim disimpan dalam Tx I Buffer dan sampel *Quadrature* yang akan dikirim disimpan dalam Tx Q Buffer
- DAC I/Q : Digital-to-Analog Converter untuk sampel In-Phase/Quadrature
- Tx BB Amplifier : Transmitter Base Band Amplifier. Ada dua amplifier, satu untuk sinyal In-Phase dan satu untuk sinyal Quadrature. Ini adalah variabel Gain amplifier yang keduanya diatur dengan nilai penguatan yang sama, yang dapat dimasukkan oleh user dari workspace matlab.
- Upconversion : mengubah sinyal base band ke sinyal RF. Frekuensi carrier tergantung pada pengaturan PLL.
- Tx RF amplifier : transmitter RF amplifier, variabel gain amplifier, nilai – nilai penguatan yang dapat dimasukkan oleh user dari workspace Matlab.

Blok Pada Jalur Penerima :

- Rx RF Amplifier : receiver RF amplifier, variabel gain amplifier, nilai penguatan yang dapat dimasukkan oleh user dari workspace di matlab
- Down conversion dari RF tergantung pada pengaturan PLL.
- Rx BB Amplifier : receiver basis band amplifier. Ada dua amplifier, satu untuk sinyal I dan satu untuk sinyal Q. Ini adalah variabel gain amplifier yang keduanya diatur dengan nilai penguatan yang sama, yang dapat dimasukkan oleh user dari workspace Matlab.
- RSSI : received signal strength indicator. Blok ini mengukur RSSI tersebut
- ADC I/Q : analog-to-digital converter untuk sampel In-phase/Quadrature
- ADC RSSI : analog to digital converter untuk pengukuran RSSI. Data RSSI tersedia di $\frac{1}{4}$ laju data I/Q
- Rx I/Q buffer: sampel In-phase yang diterima akan disimpan dalam Rx buffer I dan sampel Quadrature yang diterima akan disimpan di Rx Q buffer
- RSSI buffer : RSSI data yang disimpan dalam buffer ini

Block PLL : pengaturan PLL menentukan frekuensi pembawa yang dapat diatur ke salah satu dari 14 kanal di Wi-fi 2,4 GHz band atau 23 kanal di 5 GHz. Saluran pembawa dapat diatur langsung dari workspace Matlab.

CATATAN : receiver RF dan BB gain dapat diatur menggunakan Gain manual control (MGC) atau automatic gain control (AGC). Dalam model MGC pengguna mengambil nilai – nilai dari penguatan dan menetapkan penguatan pada modul menggunakan fungsi yang tersedia dalam referensi WARPLab M-Code. Dalam mode AGC papan dikonfigurasi sehingga secara otomatis menetapkan penguatan penerima berdasarkan pengukuran RSSI. Framework WARPLab memberikan contoh yang menggambarkan bagaimana menggunakan mode MGC dan mode AGC.

2.6.3 Gambaran Framework WARPLab

WARPLab adalah framework yang bisa menggabungkan WARP dan Matlab. Dengan WARPLab, dapat berinteraksi dengan node WARP langsung dari workspace Matlab dan sinyal yang dihasilkan

dalam MATLAB dapat ditransmisikan secara real-time melalui udara (over-the-air) menggunakan node WARP.

WARPLab sangat fleksibel, mendukung kombinasi dari perangkat keras WARP v1,v2 atau v3. Selain itu juga mendukung sejumlah radio pada setiap board WARP. Desain untuk PHY baru adalah sebagai berikut:

- Pengguna membuat dalam Matlab, sampel yang akan dikirim adalah bagian dari custom PHY.
- Sampel yang akan dikirim akan didownload ke buffer di node yang digunakan sebagai pemancar
- Pengguna mengirimkan trigger untuk node transmitter dan receiver. Setelah penerimaan trigger ini, sampel dikirim over-the-air dan di-capture secara real time
- Pengguna membaca sampel ter-capture yang diambil dari node penerima dengan workspace matlab
- Sampel yang diterima diproses secara offline di Matlab

2.6.4 Spesifikasi WARPLab

Tx/Rx I/Q Buffers

- Tx/Rx I/Q Buffers yang independent
- Setiap buffer dapat menyimpan maksimum 2^{14} sampel
- Frekuensi sampling yang digunakan adalah 40 MHz

Syarat sinyal yang dipancarkan :

- Amplitudo pada bagian real (in-phase signal: I) adalah [-1,1] dan amplitudo pada bagian imajiner (quadrature signal :Q) adalah [-1,1]. Sinyal I/Q pada pemancar terwakili dalam FPGA menggunakan 15-16 titik tetap.
- Frekuensi terendah : 30 KHz (Radio Filter DC)
- Frekuensi tertinggi : bergantung pada pengaturan frekuensi pada Tx/Rx Low Pas Filter (LPF) secara default Tx dan Rx LPF diatur oleh nominal mode. Adapun pengaturannya adalah sebagai berikut:

LPF pada pemancar:

- Mode 0 : undefined
- Mode 1 : 12 MHz (Nominal Mode)
- Mode 2 : 18 MHz (Turbo Mode 1)
- Mode 3 : 24 MHz (Turbo Mode 2)

LPF pada penerima:

- Mode 0 : 7.5 MHz
- Mode 1 : 9.5 MHz (Nominal Mode)

- Mode 2 : 14 MHz (Turbo Mode 1)
 - Mode 3 : 18 MHz (Turbo Mode 2)
 - Frekuensi sampling yang digunakan adalah 40 MHz
- I/Q ADC/AC
- Clock selalu pada 40 MHz
 - Masukan sinyal DAC I/Q direpresentasikan menggunakan titik tetap 16_15
 - Masukan sinyal ADC I/Q direpresentasikan menggunakan titik tetap 14_13

Tx/Rx Amplifiers

- Tx RF PA : 30 dB
- Tx BB : 0-3 dB dengan step 1,5 dB
- Tx RF : 0-63 dB dengan step 0.5 dB
- Rx BB : 0-31 dB dengan step 2 dB
- Rx RF : 0-63 dB

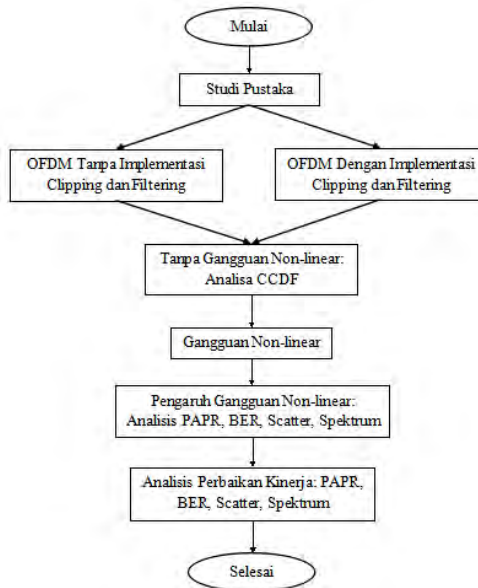
BAB 3

DESAIN DAN SIMULASI SISTEM

Pada bab ini akan dijelaskan mengenai pemodelan blok sistem “Reduksi PAPR sinyal OFDM dengan metode *clipping* dan *filtering*” sehingga dapat disimulasikan dan selanjutnya diimplementasikan pada WARP.

3.1 Alur Pengerjaan Tugas Akhir

Dalam melakukan perancangan sistem perlu dilakukan beberapa tahapan diantaranya tahap pemodelan sistem, analisa, simulasi serta implementasi sistem.



Gambar 2.1 Flowchart Pengerjaan Tugas Akhir

Tahap pertama adalah studi pustaka mengenai OFDM, teknik modulasi M-ary QAM, PAPR, teknik *clipping* dan *filtering*, distorsi nonlinier model Saleh, dan WARP. Lalu dengan menggunakan *software* Matlab yang telah diimplementasikan pada WARP, sinyal OFDM diamati dengan dan tanpa teknik reduksi *clipping* dan *filtering*. Setelah itu, dengan menyisipkan gangguan nonlinier model Saleh, diamati

PAPR pada grafik CCDF, BER, dan *scatter plot*. Lalu, dari hasil perhitungan, dapat dianalisa teknik reduksi *clipping* dan *filtering* jika diberikan distorsi nonlinier.

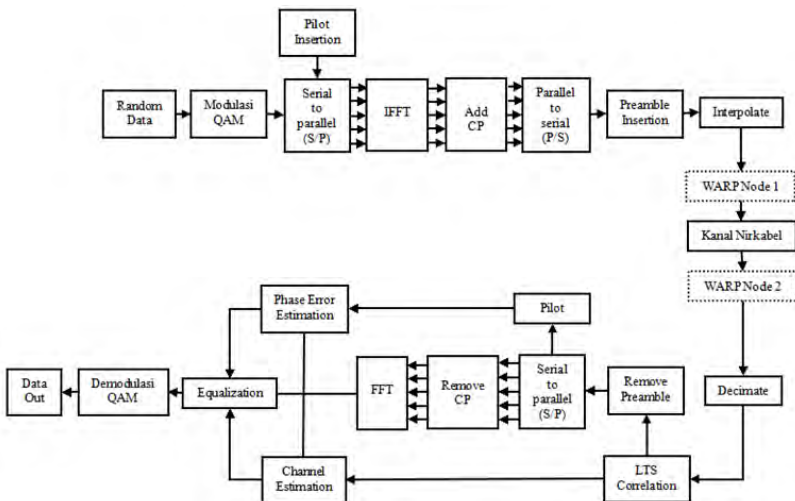
3.2 Desain Sistem OFDM

Tujuan dari proses desain sistem OFDM ini adalah membuat sebuah sistem komunikasi antar dua node WARP sesuai dengan kriteria sebagai berikut:

- Sistem bisa mengirimkan jumlah bit yang sangat besar; pada 16-QAM sejumlah 17280 bit dan pada 64-QAM sejumlah 25920 bit
- Dapat melakukan proses estimasi kanal
- Dapat melakukan proses estimasi fase *error*
- Mengurangi kemungkinan kesalahan deteksi simbol atau perbaikan dalam struktur *preamble*

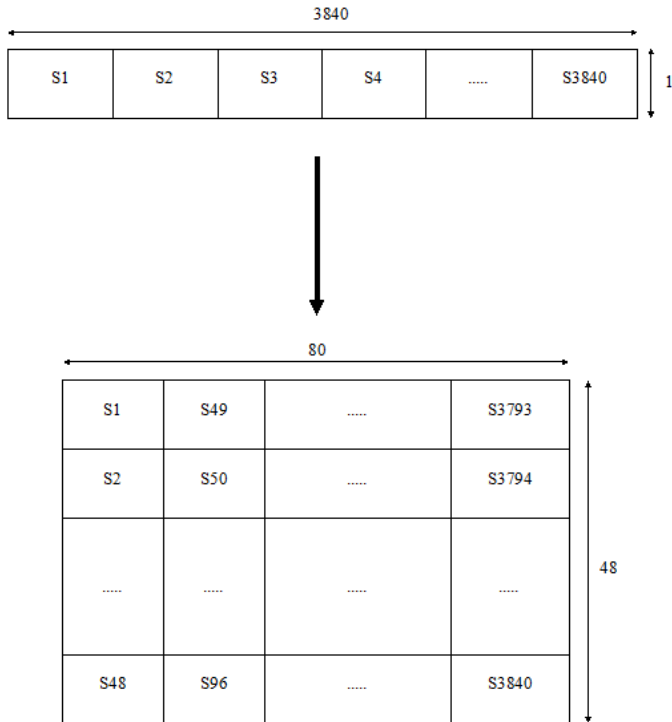
3.2.1 Sistem OFDM pada WARP

Secara garis besar, sistem OFDM pada WARP diperlihatkan pada Gambar 3.2. Data acak akan dibangkitkan, dengan jumlah sesuai pada desain yang telah dibuat. Kemudian data acak tersebut akan dimodulasi dengan skema modulasi 16-QAM dan 64-QAM.



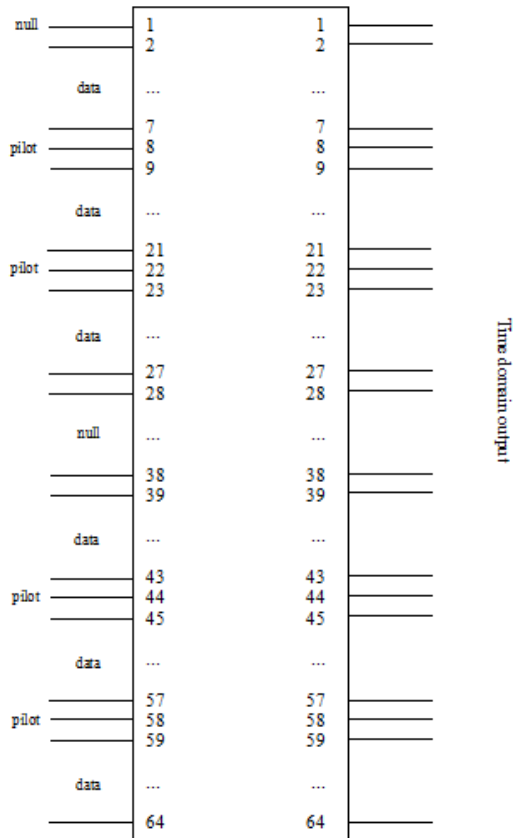
Gambar 2.2 Diagram Blok Sistem OFDM pada WARP

Setelah mengalami proses modulasi, maka data akan dibuat dalam bentuk paralel dengan ukuran sebesar 48x80. Proses pengubahan dari bentuk serial menjadi paralel diperlihatkan pada Gambar 3.3.



Gambar 2.3 Konversi Serial ke Paralel

Setelah dalam bentuk paralel, kemudian akan masuk ke dalam proses IFFT dengan ukuran 64. Keluaran dari proses IFFT yaitu matriks dengan ukuran 64x80, dari 64 baris yang ada baris yang ditempatkan untuk data yaitu 48 sementara untuk pilot yaitu 4 seperti yang diperlihatkan pada Gambar 3.4.

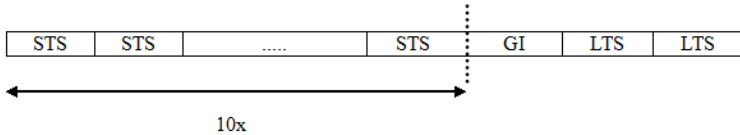


Gambar 2.4 Penempatan Data dan Pilot pada Blok IFFT

Pilot ditempatkan pada subcarrier 8, 22, 44, dan 58 untuk membantu pada saat proses *synchronization*. Setelah semua data dan pilot ditempatkan pada posisi masing-masing maka langkah selanjutnya yaitu menambahkan *cyclic prefix* dengan ukuran 25% dari satu simbol OFDM. Dalam hal ini, satu simbol OFDM memiliki ukuran panjang 64 sehingga ukuran *cyclic prefix* nya yaitu 16.

Salinan simbol untuk *cyclic prefix* dimulai pada simbol ke 64 sampai ke 80, yang kemudian dipindahkan pada bagian awal *frame*.

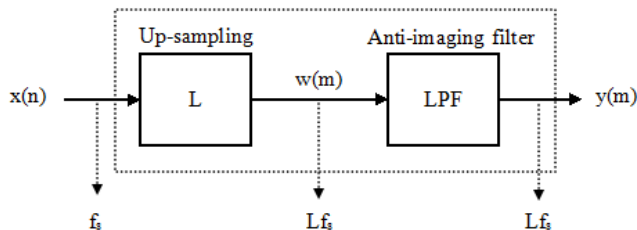
Setelah itu, maka langkah selanjutnya yaitu membuat matriks dalam bentuk paralel tersebut menjadi bentuk serial dalam hal ini ukurannya menjadi 1x6400. Kemudian menempatkan *preamble* sebelum data dengan ukuran 1x6400 tersebut. *Preamble* disini merupakan gabungan antara *Short Training Symbol* (STS) dan *Long Training Symbol* (LTS), susunan dari *preamble* ini diperlihatkan pada Gambar 3.5.



Gambar 2.5 Susunan Preamble

Susunan preamble terdiri atas 10 STS dan 2 LTS yang dipisahkan oleh GI. Tujuh STS pertama digunakan untuk deteksi sinyal, kontrol gain otomatis, dan mendeteksi carrier. Tiga STS terakhir digunakan untuk estimasi kasar dari frekuensi dan clock-off. Sedangkan LTS berguna untuk menentukan letak simbol awal dari OFDM.

Setelah menempatkan *preamble*, maka langkah selanjutnya yaitu proses interpolasi. Proses ini menjadi cukup penting bila akan diimplementasikan pada WARP, karena bila tidak menggunakan interpolasi maka dikhawatirkan frekuensi selektif pada kanal wireless akan merusak sinyal dan mengakibatkan sinyal OFDM yang kita desain tersebut lebih rentan terhadap error. Interpolasi digunakan agar diperoleh impuls respon kanal estimasi pada semua simbol OFDM yang dikirim. Proses interpolasi diperlihatkan pada Gambar 3.6 di bawah ini:



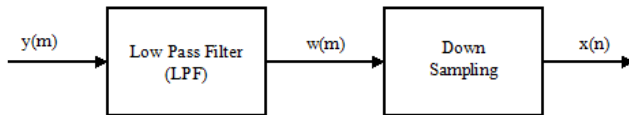
Gambar 2.6 Blok Diagram Interpolasi

Pada Gambar 3.6 dapat terlihat bahwa pada proses interpolasi terdapat dua tahap. Tahap pertama yaitu *Up-sampling* dengan ukuran L

yaitu $L > 1$ dan tahapan terakhir yaitu sinyal hasil dari interpolasi itu akan masuk ke dalam *Low Pass Filter* (LPF), di mana LPF ini merupakan respon impuls dari $h(m)$. LPF digunakan untuk memfilter sinyal dari hasil *up-sample*.

Setelah melalui proses interpolasi selesai maka sinyal akan mengalami langkah terakhir sebelum dipancarkan yaitu sinyal akan dinormalisasi untuk mengoptimalkan DAC yang terdapat pada WARP, sehingga nilai sinyal akan berada pada range $+1$ dan -1 . Kemudian sinyal tersebut dikirimkan ke *buffer transmitter* melalui *Ethernet*. Proses transmisi akan berjalan setelah dikirimkan sinyal sinkronisasi ke node Tx dan Rx.

Setelah sinyal dikirim melalui kanal *real* maka sinyal akan diterima oleh penerima. Langkah pertama yang harus dilakukan di penerima yaitu proses desimasi. Proses desimasi ini memiliki fungsi yang berlawanan dengan proses interpolasi. Proses desimasi diperlihatkan pada Gambar 3.7.



Gambar 2.7 Blok Diagram Desimasi

Pada proses desimasi ini perbedaan yang mencolok yaitu jika pada proses interpolasi ada blok diagram *up-sampling* dengan variabel L maka proses desimasi ini terdapat blok diagram *down-sampling* dengan variabel M . Namun yang harus diingat yaitu nilai L dan M harus sama.

Setelah sinyal mengalami proses desimasi, langkah selanjutnya yaitu melakukan *cross correlation* antara *preamble* yang terdapat pada penerima dengan satu LTS yang terdapat pada bagian pemancar. Rumus untuk *cross correlation* digambarkan pada persamaan:

$$C(n) = \sum_{l=0}^M \sum_{k=0}^N r(l * N \neq k \neq n) s^*(l * N \neq k) \quad (3.1)$$

Di mana:

- r : Keseluruhan *preamble* pada penerima
- s : Satu buah LTS

N : Panjang dari data OFDM (64 sampel)
 M : Jumlah dari LTS yang di *cross correlation*

Proses korelasi ini bertujuan untuk 2 hal yaitu melakukan estimasi kanal dan juga menentukan awal dari *frame* OFDM yang pertama. Proses korelasi ini terjadi antara sinyal yang diterima dengan deretan 1 buah LTS yang terdapat pada pemancar sehingga akan didapatkan dua nilai puncak hasil korelasi dengan LTS dikarenakan pada *preamble* terdapat dua buah LTS. Nilai puncak yang terakhir dari hasil korelasi tersebut yang akan digunakan untuk mendapatkan frame OFDM yang pertama.

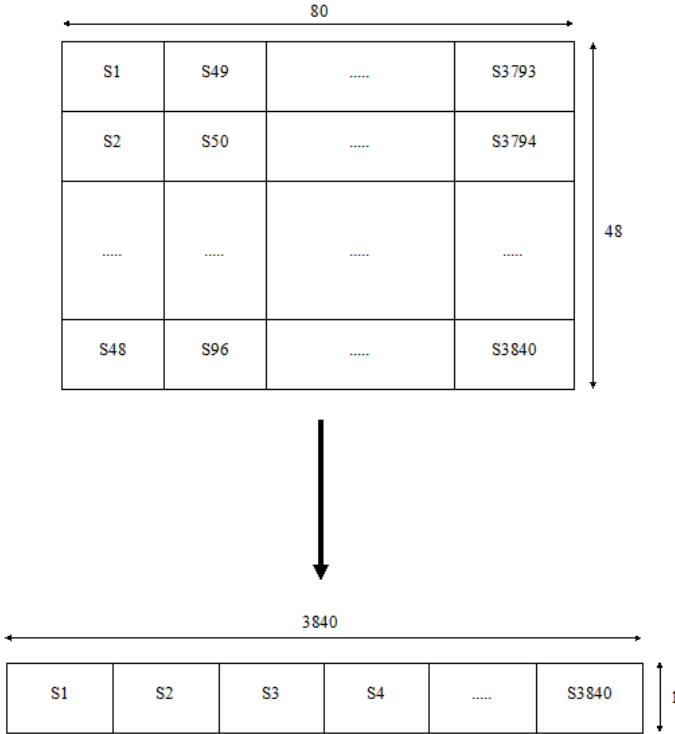
Kemudian setelah mendapatkan hasil korelasi maka kita akan mendapatkan *frame* awal OFDM. Bentuk *frame* OFDM yang sudah didapat tersebut diubah menjadi dalam bentuk paralel.. Setelah itu *cyclic prefix* yang tadi disisipkan pada proses pemancar dihilangkan pada proses ini. Setelah *cyclic prefix* dihilangkan langkah selanjutnya yaitu proses FFT dengan ukuran yang sama seperti pada pemancar yaitu 64. Dalam hal ini, sinyal yang dihasilkan sudah dalam domain frekuensi.

Setelah melalui proses FFT, sinyal tersebut akan mengalami proses *equalizer* dengan kanal estimasi yang didapat dari proses korelasi LTS dan *error phase* dengan menggunakan pilot. Data informasi tersebut akan diubah menjadi dalam bentuk serial sebelum memasuki proses demodulasi.

Simbol yang berjumlah 80 terdiri atas data yang berjumlah 48 di tiap simbolnya. Setelah diubah ke bentuk serial, maka data akan berjumlah 48x80 atau sebanyak 3840. Setelah data berupa serial, maka data dapat masuk ke DAC sehingga siap ditransmisikan, karena data harus berupa analog untuk dipancarkan.

Setelah bentuk data informasi dalam bentuk serial, maka langkah selanjutnya yaitu proses demodulasi. Proses demodulasi ini bertujuan untuk mengubah simbol menjadi bentuk bit informasi. Setelah bit-bit hasil demodulasi didapat, maka akan dibandingkan *error* yang terdapat antara bit input dengan bit output yang dinyatakan dalam bentuk rasio/*Bit Error Rate* (BER).

Gambar 3.8 menunjukkan proses paralel ke serial pada OFDM, yang semula data berbentuk 48x80 menjadi data serial dengan bentuk 3840x1.



Gambar 2.8 Konversi Paralel ke Serial

3.2.2 Estimasi Kanal

Estimasi kanal adalah suatu teknik yang digunakan pada proses transmisi yang bertujuan untuk mengestimasi impuls repon dari suatu kanal terhadap sinyal terkirim. Efek perubahan terhadap sinyal terkirim yang dihasilkan oleh kanal harus dilakukan estimasi agar pendeteksian sinyal informasi menjadi lebih akurat.

Sebuah LTS pada penerima yang sudah melalui proses blok FFT dinotasikan dengan $R_{l,k}$ ($l=0.1$) digambarkan dalam bentuk di mana $W_{l,k}$ ditambah dengan hasil perkalian dari LTS pada pemancar $d_{l,k}$ dan kanal $H_{l,k}$.

$$R_{l,k} = H_{l,k} d_{l,k} + W_{l,k} \quad (3.2)$$

Persamaan estimasi kanal menggunakan LTS menggunakan persamaan:

$$\begin{aligned}
 H_k &= \frac{1}{2} (R_{0,k} + R_{1,k}) \frac{1}{d_k} \\
 &= \frac{1}{2} (H_k d_k + W_{0,k} + H_k d_k + W_{1,k}) \frac{1}{d_k} \\
 &= H_k + \frac{1}{d_k} \frac{(W_{0,k} + W_{1,k})}{2}
 \end{aligned} \tag{3.3}$$

Setelah nilai dari estimasi kanal didapat, maka dilakukan proses ekualisasi terhadap *payload* (total keseluruhan simbol OFDM) yang diterima dengan menggunakan persamaan:

$$x_e(k) = \frac{x(k)}{H_k} \tag{3.4}$$

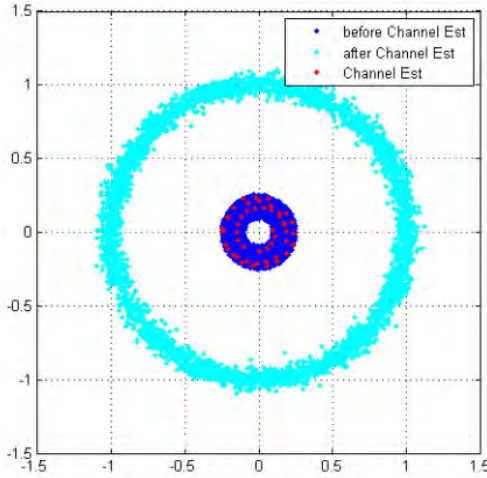
Di mana:

$x_e(k)$: Sinyal setelah melalui proses ekualisasi
$x(k)$: Sinyal sebelum melalui proses ekualisasi
H_k	: Nilai estimasi kanal

Perubahan nilai saat sebelum melalui estimasi dan sesudah estimasi dalam bentuk konstelasi sangat jelas terlihat seperti pada Gambar 3.9. Jika dilihat pada Gambar tersebut, fungsi dari estimasi kanal ini yaitu untuk mengembalikan bentuk sinyal ke lingkaran aslinya.

3.2.3 Estimasi Fasa Error

Setelah proses estimasi kanal selesai, maka langkah selanjutnya yaitu melakukan koreksi fasa *error*. Estimasi ini diperlukan karena meskipun sinyal sudah berada di dalam lingkaran yang tepat namun masih terdapat sisa (residual) *error* pada *timing offset* yang disebabkan oleh adanya karakteristik variasi waktu pada kanal. Dampak dari *error* ini yaitu ketidaktepatan letak simbol-simbol OFDM.



Gambar 2.9 Diagram Konstelasi Estimasi Kanal

Seperti yang dijelaskan sebelumnya, terdapat 4 buah pilot yang terdapat pada masing-masing simbol OFDM, pilot-pilot tersebut yang akan digunakan untuk mengurangi sisa error pada timing offset tersebut. Penempatan letak pilot tersebut bertujuan untuk menghindari pilot yang akan dikirim terganggu satu sama lain.

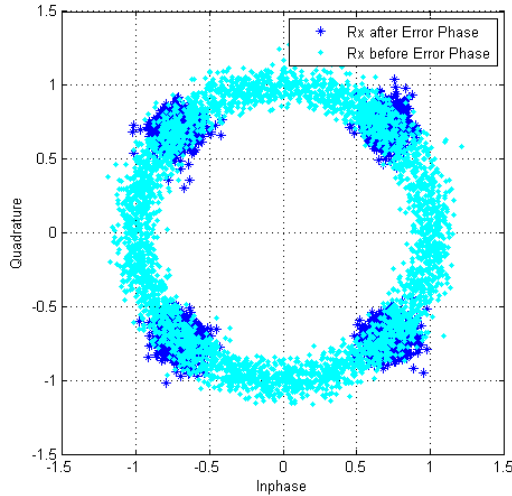
Proses estimasi fasa error cukup simpel yaitu dengan mengalikan simbol yang diterima dengan $e^{j\theta}$ dimana θ adalah sudut dari estimasi fasa. Secara matematis, operasi perbaikan fasa dilakukan dengan persamaan sebagai berikut:

$$s'(k) = s(k)e^{-j\theta} \quad (3.5)$$

Di mana:

- $s'(k)$: Nilai sinyal diterima setelah proses koreksi fasa
- $s(k)$: Nilai sinyal diterima sebelum proses koreksi fasa
- θ : Sudut dari estimasi fasa

Nilai θ didapat dari rata-rata perbedaan nilai sudut antara pilot yang diterima dengan pilot yang dipancarkan. Efek dari penggunaan koreksi fasa *error* bisa dilihat pada Gambar 3.10.

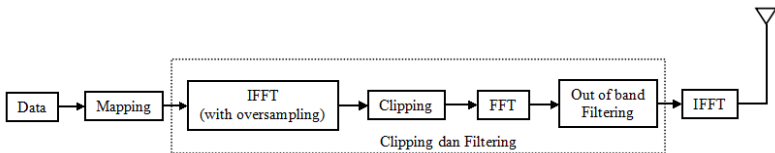


Gambar 2.10 Perbandingan Fasa Sinyal dengan Koreksi Fasa

3.3 Metode Reduksi PAPR (Clipping dan Filtering)

Metode yang digunakan untuk mereduksi PAPR pada sistem ini adalah *clipping* dan *filtering*, yaitu pencacatan sinyal dengan cara memotong sinyal puncak. Level *clipping* yang digunakan dalam Tugas Akhir ini sebesar 0.95, 0.90, 0.85, 0.80, dan 0.75.

Dalam simulasi, sinyal OFDM yang telah dibangkitkan akan ditransmisikan melewati kanal AWGN. Blok “AWGN channel” adalah sebuah blok yang berfungsi untuk mensimulasikan kanal agar sesuai dengan kanal yang sebenarnya di mana terdapat *noise* yang berupa AWGN (*Additive White Gaussian Noise*) dan diberikan distorsi nonlinier yang menggunakan model Saleh. Parameter yang diubah-ubah dalam penentuan nilai PAPR pada simulasi ini adalah level clipping 0.95, 0.90, 0.85, 0.80 dan 0.75 dan Signal to Noise Ratio (SNR) dari 0-60 dB dengan selisih 10 dB.



Gambar 2.11 Blok Diagram Clipping Filtering pada Sistem OFDM

Pada pengukuran oleh WARP, kanal AWGN tidak digunakan karena antara pemancar dan penerima terdapat kanal dalam bentuk *real*, distorsi nonlinierpun dihilangkan karena distorsi dihasilkan dari *gain* pemancar yang nilainya diubah-ubah lalu akan diamati perubahan *Bit Error Rate* (BER) dari *gain* 50 sampai dengan *gain* 60, dengan selisih sebesar 2 dB.

3.4 Perhitungan PAPR

Peak to Average Power Ratio (PAPR) dihitung per simbol OFDM, di mana dalam satu simbol OFDM terdiri dari 16 bit data. Karena itu sinyal informasi yang dikirim harus *buffer* terlebih dahulu tiap 16 bit lalu *direshape* untuk menyamakan matriksnya. Kemudian dihitung PAPR dengan menggunakan rumus sesuai persamaan. PAPR dihitung saat sinyal sebelum dan sesudah *clipping*. Kemudian kedua nilai PAPR tersebut dibandingkan dan dianalisa.

Perhitungan PAPR per simbol OFDM diperoleh dari kuadrat daya maksimal dibagi kuadrat daya rata-rata. Setelah diperoleh nilai PAPR per simbol OFDM, maka dihitung nilai PAPR rata-rata dari semua simbol OFDM yang dibangkitkan. *Script* perhitungan nilai PAPR menggunakan *software* Matlab dapat dilihat pada Gambar 3.12.

```

1  %ofdm_symbol=ifft_in_mat;
2  ofdm_symbol=ifft_in_mat;
3  time_domain_signal=(ifft(ofdm_symbol,64));
4  meano=mean(abs(time_domain_signal).^2);
5  peako=max(abs(time_domain_signal).^2);
6  papro=10*log10(peako./meano);
7  x=reshape(time_domain_signal,1,[]);
8  z=x;
9  PersenA=0.55;%Nilai PersenA bisa diubah
10 A=PersenA*mean(max(abs(time_domain_signal)));
11 y=zeros(size(x));
12 for k=1:length(x)
13     if abs(x(k)) <=A
14         y(k)=x(k);
15     else
16         y(k)=A*exp(1i*angle(x(k)));
17     end
18 end
19 signal_Tx=reshape(y,N_SC,[]);
20
21 meanz=mean(abs(signal_Tx).^2);
22 peakz=max(abs(signal_Tx).^2);
23 paprz=10*log10(peakz./meanz);

```

Gambar 2.12 Script Matlab Perhitungan PAPR

Di mana:

A : level *clipping*
papro : nilai PAPR awal
paprz : nilai PAPR setelah *diclipping*

3.5 Perhitungan BER

Apabila nilai bit sinyal output tidak sama dengan nilai bit sinyal input yang dibangkitkan berarti terjadi kesalahan bit. Untuk membandingkan antara bit data input dan output maka dihitung nilai *Bit Error Rate* (BER). Nilai BER, jumlah *error*, dan jumlah bit yang dikirim akan ditampilkan pada figure Matlab.

3.6 Implementasi Sistem pada WARP

Perangkat lunak yang digunakan dalam Tugas Akhir ini adalah:

- Matlab
- TCP/IP Toolbox
- WARPLab

3.6.1 Matlab R2014a

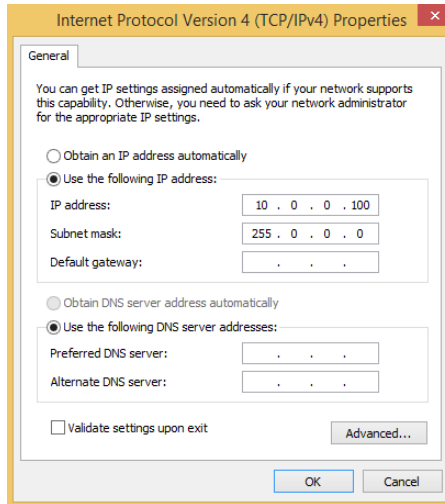
MATLAB adalah singkatan dari MATrix LABoratory, merupakan bahasa pemrograman yang dikembangkan oleh The Mathwork Inc. berbasis matriks dan digunakan untuk analisis dan komputasi numerik, plotting fungsi dan data, implementasi algoritma, pembuatan interface pengguna, dan interface dengan program dalam bahasa lainnya. Software MATLAB yang digunakan adalah MATLAB versi R2014a.

Script MATLAB yang digunakan pada penelitian ini telah terlampir. PC yang terdapat program MATLAB akan dihubungkan pada WARP melalui *Ethernet* menggunakan kabel LAN. Setelah sinyal ditransmisikan dan diterima oleh antena pada WARP, selanjutnya akan dikembalikan ke MATLAB untuk dilakukan perhitungan matematisnya.

3.6.2 Setting TCP/IP

Setelah matlab diinstal, buka CMD yang ada di PC dan lakukan pengaturan sebagai berikut:

Untuk WARP : 10.0.0.1 dan 10.0.0.2 atau lainnya
Untuk CPU : 10.0.0.99 atau lainnya
Subnetting : 255.255.255.0



Gambar 2.13 Pengaturan IP Address untuk WARP

Pengaturan IP address untuk WARP diperlihatkan pada Gambar 3.13. Untuk test ping ke WARP dan cek ARP table di CPU menggunakan command “arp-a” untuk ARP table dan “ping 10.0.0.1” untuk ping. Jika muncul reply time, maka telah berhasil.

Cara mengatur TCP/IP yaitu:

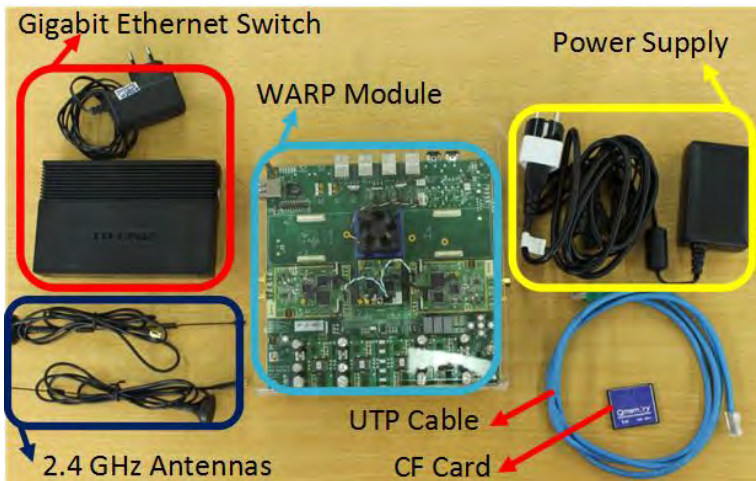
1. Masuk ke program Matlab
2. Pilih set Path
3. Klik add folder
4. Pilih folder tcp_udp_ip

3.6.3 Kelengkapan Perangkat Keras

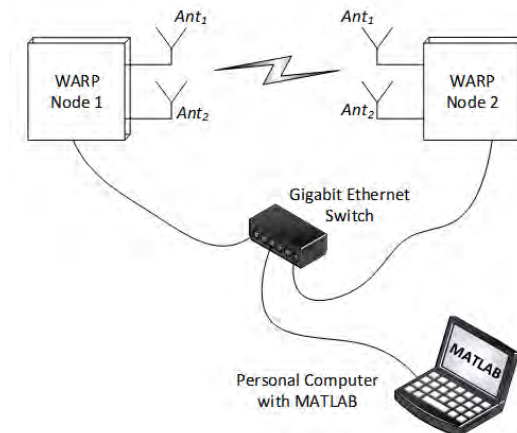
Komponen-komponen WARP yang dibutuhkan dalam pengukuran diperlihatkan oleh Gambar 3.14. Selain WARP, *hardware* lain yang dibutuhkan adalah komputer/PC, di mana dalam pengukuran ini menggunakan komputer WARP *Research Team* yang ada di Laboratorium Komunikasi Multimedia B304.

WARP Module terdapat 2 buah, yaitu *transmitter* dan *receiver*. Gigabit ethernet switch digunakan untuk menghubungkan perangkat WARP dan PC dengan menggunakan kabel UTP. Antena yang digunakan dalam pengukuran yaitu antena 2.4 GHz.

Konfigurasi perangkat pengukuran diperlihatkan pada Gambar 3.15. PC yang telah terdapat Matlab di dalamnya akan dihubungkan dengan WARP melalui Ethernet dengan kabel LAN.



Gambar 2.14 Kelengkapan Perangkat WARP



Gambar 2.15 Konfigurasi Perangkat Pengukuran

3.6.4 Langkah-langkah Konfigurasi Awal Program pada WARP

Sebelum WARP digunakan, sebelumnya kita harus mengatur konfigurasi awal dari WARP itu sendiri. Langkah-langkahnya yaitu sebagai berikut:

1. Masuk program Matlab
2. Pilih folder WARPLab dan pilih file M_code_reference di folder WARPLab_reference_design_7.10
3. Ketika ada file wl_setup.m, klik kanan dan pilih RUN
4. Setting IP sesuai IP komputer yang digunakan kemudian tekan enter sampai muncul “setup complete”
5. Masukkan command “wl_initNodes(2)” sehingga muncul “Displaying properties of 2 wl_node objects”

3.7 Skenario Pengukuran

Pada bagian ini akan dibahas tentang pengukuran pada lingkungan *indoor*. Proses pengukuran untuk analisa secara umum dijabarkan pada poin-poin berikut:

- Analisa sistem komunikasi OFDM
- Pengukuran kondisi *line-of-sight* dengan variasi *gain* dan *clipping factor*

Dalam pengukuran ini, kinerja yang diukur yaitu *Bit Error Rate* (BER) dan grafik CCDF dari *Peak to Average Power Ratio* (PAPR) dari setiap pengukuran.

Pengukuran pada sub bab ini bertujuan untuk mengetahui kinerja sistem OFDM. Hal ini diperlukan untuk mengetahui bahwa OFDM mampu mengirimkan informasi bit dalam kondisi ideal.

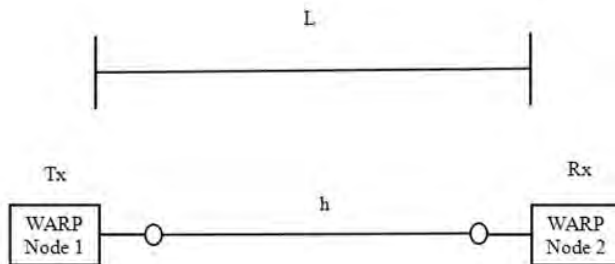
Kondisi ideal digambarkan sebagai kondisi *Line-of-sight* antara penerima dan pemancar dengan daya pancar dan gain berada pada level optimum. Level optimum terjadi ketika data yang dikirim dan diterima tidak mengalami *error*.

Kriteria pengukuran yang dilakukan dan parameter *gain setting* pada modul WARP antara lain:

- Jumlah bit yang dikirim : 17280 dan 25920
- Rx baseband gain : 2
- Rx radio frequency gain : 2

Hasil yang diharapkan dari pengukuran ini adalah memastikan bahwa desain sistem OFDM mampu mengirimkan data pada kondisi ideal.

Pengukuran dalam tugas akhir ini berada di lingkungan *indoor* dan dilakukan di dalam lab B304 dengan jarak antara pemancar dan penerima sebesar 3 meter dengan kondisi LOS/tidak terdapat *obstacle* di antara keduanya. Sketsa pengukuran diperlihatkan pada Gambar 3.16.



Gambar 2.16 Sketsa pengukuran indoor

Pada gambar di atas, L menunjukkan jarak antara WARP 1 dan 2 sedangkan h menunjukkan jarak kedua antena yang digunakan dalam pengukuran. Hasil yang diharapkan dalam pengukuran ini yaitu mengetahui perbedaan nilai BER dengan parameter nilai gain pemancar yang diubah-ubah, dari gain 50 dB hingga 60 dB dengan selisih antargain sebesar 2 dB.

BAB 4

PENGUKURAN DAN ANALISA DATA

Pada bab ini dibahas tentang analisa hasil simulasi dan implementasi reduksi PAPR pada sinyal OFDM menggunakan teknik *clipping* dan *filtering*. Ada 3 hal utama yang akan dianalisa pada bab ini, yaitu:

1. Pengaruh *Clipping Ratio* (CR)

Analisa pengaruh CR diamati dari simulasi dan akan dianalisa nilai BER dan PAPR dari setiap *Clipping Ratio*.

2. Analisa *Bit Error Rate* (BER)

Dalam menganalisa BER, akan dibandingkan OFDM dan teknik reduksi. Parameter yang diubah-ubah adalah gain pemancar di pengukuran pada WARP dan SNR pada simulasi.

3. Analisa reduksi PAPR

Dalam menganalisa reduksi PAPR, akan dibandingkan hasil reduksi teknik *clipping* dan *filtering* dengan sistem OFDM pada simulasi. Parameter yang diubah-ubah adalah SNR.

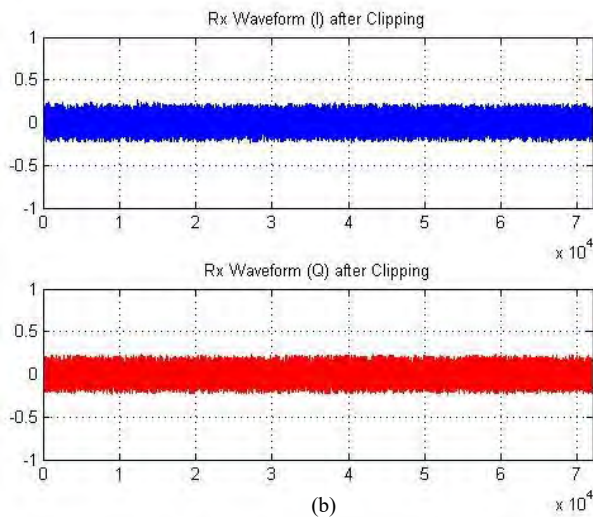
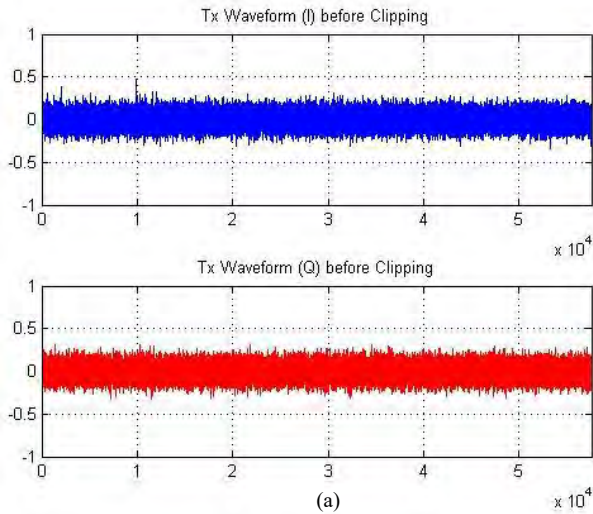
4.1 Sinyal yang Dikirim dan Diterima

Pada simulasi dan pengukuran pada WARP, diamati sinyal OFDM tanpa *clipping* dan *filtering* dan ketika metode *clipping* dan *filtering* digunakan dengan *Clipping Ratio* (CR) 0.95, 0.90, 0.85, 0.80, dan 0.75. Untuk simulasi, parameter-parameter yang digunakan ditunjukkan dalam tabel berikut ini:

Tabel 0.1 Parameter Simulasi Reduksi PAPR pada OFDM

Parameter	Nilai
Modulasi	16-QAM, 64-QAM
Model kanal	AWGN
Model distorsi	Model Saleh
Ukuran FFT	64
Jumlah Subcarrier	64
Clipping Ratio (CR)	0.95, 0.90, 0.85, 0.80, 0.75

Sinyal OFDM tanpa reduksi dan sinyal yang telah diberikan metode *clipping* dan *filtering* pada 5 *clipping ratio* 0.95, 0.90, 0.85, 0.80, dan 0.75. Salah satunya ditunjukkan oleh Gambar 4.1.



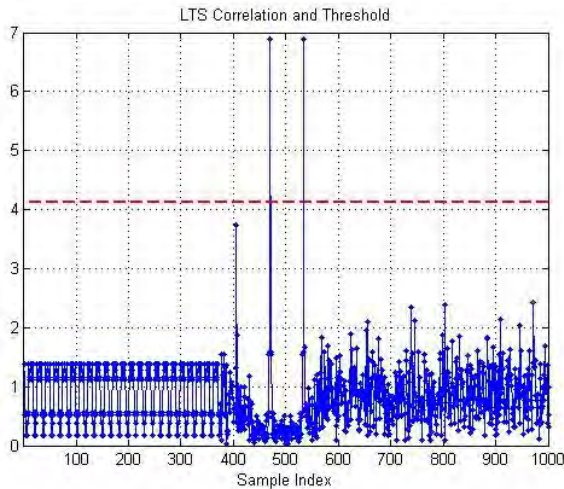
Gambar 0.1 (a) Sinyal sebelum dipotong (b) Sinyal yang dipotong dengan $CR=0.85$

Dari gambar tersebut dapat diketahui bahwa semakin rendah nilai CR, maka sinyal yang dipotong akan semakin banyak. CR 0.85 artinya yaitu

15% dari sinyal dipotong. Pengaruh pemotongan sinyal tersebut akan diamati pada bagian selanjutnya, yaitu perubahan BER dan PAPR.

4.2 Korelasi LTS

Long Training Symbol (LTS) digunakan untuk menentukan letak simbol awal dari OFDM. Gambar 4.2 menunjukkan hasil korelasi LTS yang diperoleh dari program Matlab.



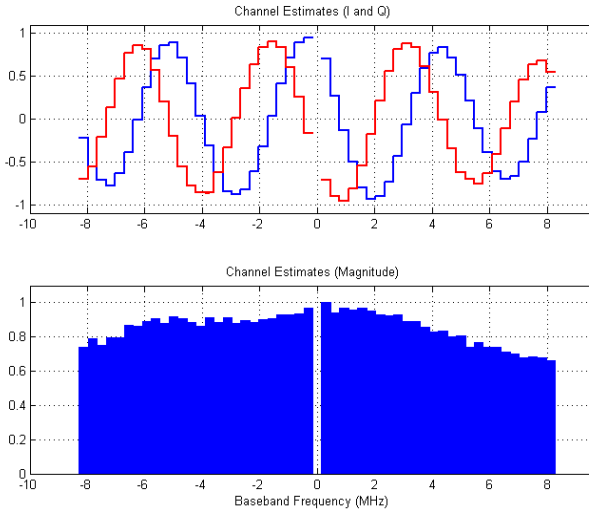
Gambar 0.2 Hasil Korelasi LTS

Pada gambar tersebut terlihat bahwa ada 2 nilai puncak yang dihasilkan dari korelasi LTS. Sementara garis merah menunjukkan nilai *threshold* yang diijinkan. Nilai *threshold* yang diijinkan pada percobaan ini yaitu sebesar 0.6 dari nilai puncak.

Dua nilai puncak tersebut menandakan lokasi dari 2 buah nilai LTS yang digunakan dalam sistem tersebut. Nilai puncak tersebut berada pada *sample* ke 470 dan 534, jika diperhatikan dari nilai kedua puncak tersebut maka didapat bahwa selisih dari kedua nilai tersebut yaitu 64 sesuai dengan banyaknya *sample* dalam 1 buah LTS. Perlu dipahami bahwa kedua nilai puncak tersebut berbeda-beda pada setiap pengukuran. Kemudian dari kedua nilai tersebut dapat diketahui letak simbol OFDM yang pertama.

4.3 Estimasi

Proses estimasi terdiri dari estimasi kanal dan fasa *error* di mana keduanya memiliki fungsi yang berbeda. Untuk nilai estimasi kanal baik sinyal *inphase* dan *quadrature* dan kompleks *magnitude* diperlihatkan pada Gambar 4.3.

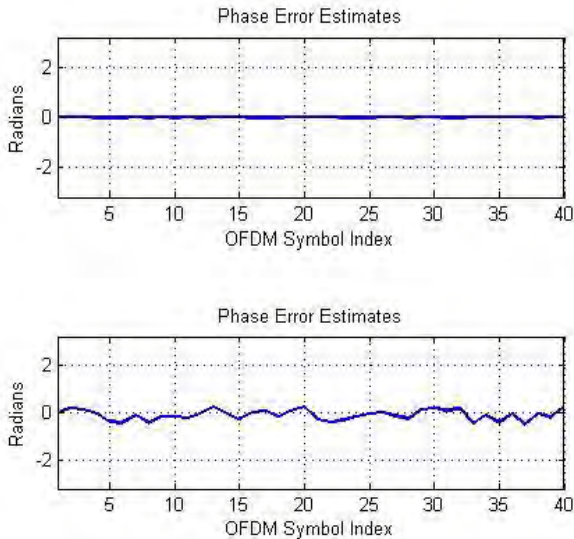


Gambar 0.3 Estimasi kanal (a) Inphase & Quadrature (b) Magnitude

Gambar 4.3 memperlihatkan nilai estimasi kanal pada setiap *subcarrier* baik nilai *inphase*, *quadrature*, maupun nilai *magnitude* dalam domain frekuensi. Dapat disimpulkan bahwa nilai estimasi pada setiap *subcarrier* berbeda-beda meskipun tidak terlalu terlihat. Nilai estimasi ini tergantung dari berapa penurunan nilai *magnitude* setelah melewati kanal pada setiap *subcarrier*.

Gambar 4.4 membandingkan nilai perubahan fasa dengan total jumlah simbol OFDM. Saat simbol OFDM yang pertama, nilai fasa *error* sangat kecil sekali jika dibandingkan dengan simbol OFDM selanjutnya. Hal ini menunjukkan bahwa posisi simbol OFDM yang pertama mendekati posisi referensi yang terdapat pada pemancar, sehingga perubahannya kecil. Begitu juga saat nilai fasa *error* semakin besar menandakan bahwa posisi simbol OFDM tersebut

semakin menjauhi posisi referensi. Hal itulah yang dinamakan *offset phase* antara pemancar dan penerima.

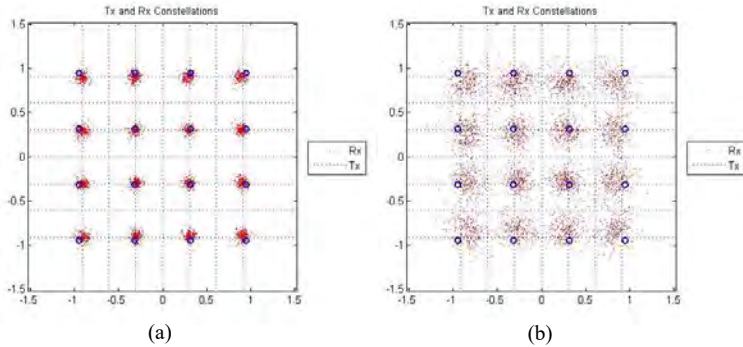


Gambar 0.4 Estimasi Error Fase pada (a) kondisi baik (b) kondisi buruk

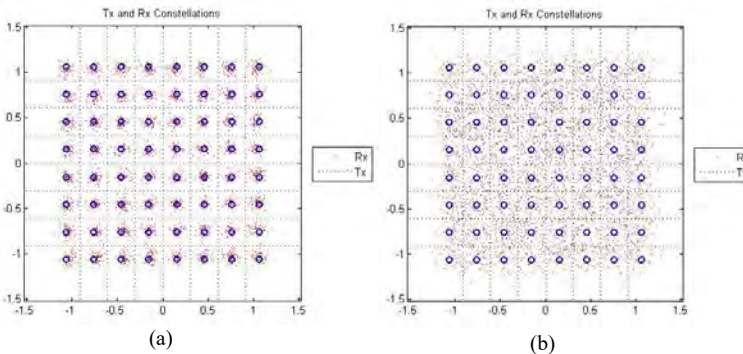
4.4 Scatterplot Sinyal

Scatterplot berfungsi untuk memberikan gambaran tentang wilayah deteksi simbol pada sistem komunikasi 16QAM dan 64QAM. Bentuk *scatterplot* yang akan ditampilkan pada gambar 4.5 dan gambar 4.6 merupakan perbandingan antara simbol yang diterima dan dipancarkan. Simbol yang diterima ini telah melalui proses estimasi kanal dan fasa *error*.

Dalam gambar tersebut dapat dilihat bahwa pada gain 50 pada modulasi 16QAM dan 64QAM, kedudukan simbol yang diterima mendekati posisi simbol yang dikirim dengan kata lain mendekati *decision area*. Sehingga pada saat pembacaan simbol yang diterima akan sama dengan simbol yang dikirim yang berarti nilai BER yang dihasilkan yaitu 0.



Gambar 0.5 Diagram Konstelasi 16QAM pada (a) Gain 50 (b) Gain 60



Gambar 0.6 Diagram Konstelasi 64QAM pada (a) Gain 50 (b) Gain 60

Setelah *gain* dinaikkan menjadi 60 dB, hasil konstelasi menunjukkan letak simbol yang diterima berada jauh dengan *decision area*. Akibat dari hal tersebut yaitu nilai BER akan naik. Analisa nilai BER akan dibahas pada bagian selanjutnya.

4.5 Pengaruh Clipping Ratio (CR) Terhadap Nilai BER dan PAPR

Simulasi pertama adalah menghitung nilai BER dan PAPR untuk mengetahui pengaruh *Clipping Ratio* (CR). Jumlah bit data yang digunakan yaitu sebesar 172800 untuk modulasi 16 QAM dan 259200 untuk modulasi 64 QAM. Jumlah *subcarrier* yaitu 64 dengan CR yang digunakan adalah 0.95, 0.90, 0.85, dan 0.75.

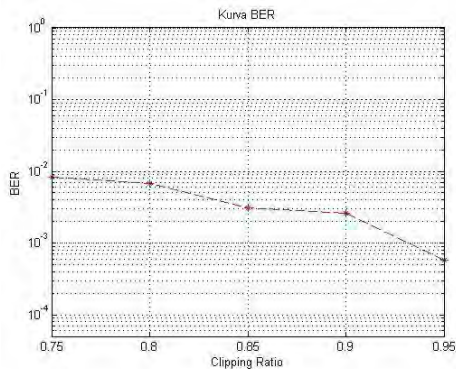
4.5.1 Analisa Perbandingan BER

Clipping ratio menentukan seberapa banyak sinyal akan dipotong. Akibat dari pemotongan sinyal pada sistem OFDM adalah sinyal menjadi cacat sehingga nilai BER bertambah. Semakin banyak amplitudo sinyal yang dipotong, maka semakin cacat sinyal tersebut sehingga BER semakin besar pula. Tabel 4.2 di bawah ini menunjukkan pertambahan nilai BER ketika *clipping ratio* turun atau ketika sinyal semakin banyak dipotong.

Tabel 0.2 Nilai BER pada Simulasi dengan Modulasi 16QAM

Clipping Ratio (CR)	Nilai BER
0.95	0.000578704
0.90	0.002627315
0.85	0.003072917
0.80	0.006863426
0.75	0.008200231

Pada saat *clipping ratio* bernilai 0.95, atau berarti 5% sinyal dipotong, nilai BER yang dihasilkan yaitu 0.000578704, atau terdapat 10 bit error dari total 17280 bit yang dikirimkan. Sinyal dipotong dengan batasan di atas sinyal rata-rata. Semakin banyak sinyal yang dipotong, maka BER yang dihasilkan akan semakin naik. Pada CR = 0.75, nilai BER mencapai 0.008200231 atau bit yang error yaitu sebanyak 141 dari total 17280 bit yang dikirim.



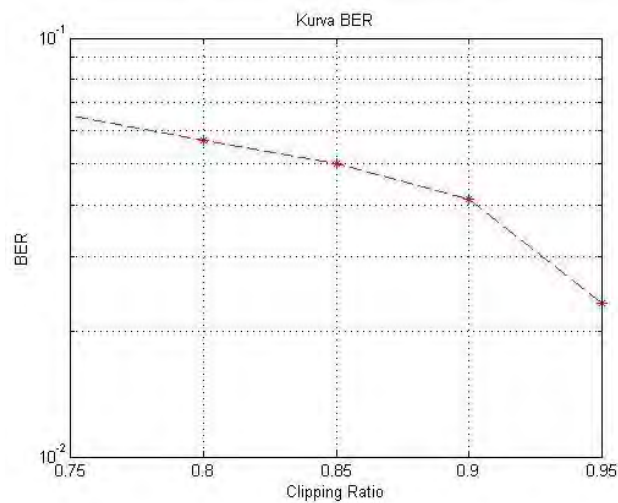
Gambar 0.7 Kurva Perbandingan BER dengan Peubah CR pada Modulasi 16QAM

Untuk modulasi 64QAM, digunakan parameter yang sama dengan parameter yang digunakan untuk simulasi pengukuran BER pada modulasi 64QAM. Namun, jumlah bit data pada 64QAM lebih banyak, yaitu sejumlah 259200 bit. Nilai BER yang diperoleh akibat perubahan nilai *clipping ratio* akan ditunjukkan pada Tabel 4.3 dan Gambar 4.12 untuk melihat penurunan BER yang terjadi pada modulasi 64QAM.

Tabel 0.3 Nilai BER pada Simulasi dengan Modulasi 64QAM

Clipping Ratio (CR)	Nilai BER
0.95	0.023260031
0.90	0.041134259
0.85	0.050243056
0.80	0.057021605
0.75	0.065204475

Nilai BER yang dihasilkan oleh modulasi 64 QAM nilainya lebih tinggi jika dibandingkan dengan nilai BER oleh modulasi 16 QAM. Hal tersebut dikarenakan jumlah bit yang dikirim jauh lebih banyak sehingga kemungkinan bit-bit *error* lebih tinggi.



Gambar 0.8 Kurva Perbandingan BER dengan Peubah CR pada Modulasi 64QAM

Nilai BER yang dihasilkan oleh modulasi 64QAM juga akan meningkat apabila nilai *clipping ratio* diturunkan atau sinyal banyak dipotong. Semakin banyak sinyal yang dipotong, maka sinyal akan menjadi cacat dan akan menyebabkan nilai BER naik.

4.5.2 Analisa Perbandingan PAPR

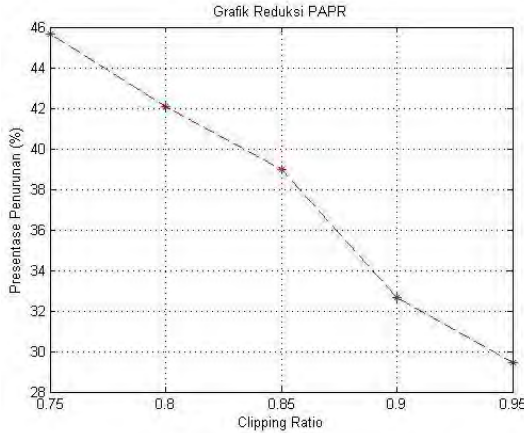
Teknik *clipping* pada OFDM bekerja dengan memotong sinyal sesuai level tertentu sehingga nilai amplitudo puncak dan amplitudo rata-rata pada sinyal nilainya akan berkurang dan bentuk sinyal menjadi lebih rata. Tabel 4.4 menunjukkan pengaruh nilai *clipping ratio* pada nilai PAPR sinyal OFDM. Data pada tabel didapat dari simulasi menggunakan teknik *clipping filtering* pada modulasi 16QAM dan mengambil nilai PAPR pada saat SNR 60 dB.

Semakin rendah nilai *clipping ratio*, maka semakin banyak sinyal yang dipotong sehingga sinyal yang tadinya memiliki nilai puncak yang tinggi nilainya akan terpotong sehingga bentuk sinyal akan semakin rata dan nilai PAPR pun akan berkurang.

Tabel 0.4 Reduksi PAPR untuk Modulasi 16QAM

Clipping Ratio (CR)	PAPR Rata-rata OFDM	PAPR Rata-rata dengan Reduksi	Selisih Penurunan	Presentase Penurunan (%)
0.95	10.339	7.296	3.043	29.43%
0.9	10.246	6.902	3.344	32.64%
0.85	10.643	6.469	4.174	39%
0.8	10.55	6.111	4.439	42.07%
0.75	10.28	5.587	4.693	45.65%

Berdasarkan data pada Tabel 4.4, perubahan *clipping ratio* dari sinyal OFDM yang akan dikirim sangat memengaruhi penurunan nilai PAPR. Grafik penurunan reduksi PAPR terhadap *clipping ratio* dapat dilihat pada Gambar 4.9 berikut ini.



Gambar 0.9 Grafik Penurunan PAPR Akibat Pengaruh CR pada Modulasi 16QAM

Berdasarkan grafik pada Gambar 4.9, dapat diketahui bahwa semakin kecil nilai *clipping ratio*, reduksi PAPR akan semakin besar, di mana pada saat $CR=0.95$, reduksi PAPR yang dihasilkan sebesar 29.43% sedangkan saat $CR=0.75$, reduksi PAPR yang dihasilkan mencapai 45.65%.

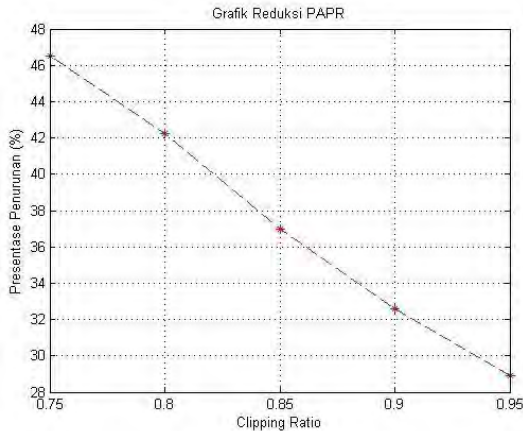
Sedangkan pada Tabel 4.5 menunjukkan pengaruh nilai *clipping ratio* pada nilai PAPR sinyal OFDM pada modulasi 64QAM dari simulasi menggunakan teknik *clipping filtering* dan mengambil nilai PAPR pada saat SNR 60 dB.

Tabel 0.5 Reduksi PAPR untuk Modulasi 16QAM

Clipping Ratio (CR)	PAPR Rata-Rata OFDM	PAPR Rata-Rata dengan Reduksi	Selisih Penurunan	Presentase Penurunan (%)
0.95	10.375	7.376	2.999	28.90%
0.9	10.338	6.968	3.37	32.60%
0.85	10.377	6.534	3.843	37%
0.8	10.622	6.134	4.488	42.25%
0.75	10.39	5.559	4.831	46.50%

Berdasarkan data pada Tabel 4.5, perubahan *clipping ratio* dari sinyal OFDM yang akan dikirim sangat memengaruhi penurunan nilai PAPR. Grafik penurunan reduksi PAPR terhadap *clipping ratio* dapat dilihat pada Gambar 4.10.

Pada saat $CR = 0.95$, presentase penurunan nilai PAPR bernilai 28.90%, dari yang awalnya bernilai 10.375 menurun menjadi 7.376 atau terjadi penurunan sebesar 2.999. Semakin rendah nilai *clipping ratio*, maka nilai PAPR pun semakin menurun. Penurunan PAPR pada saat $CR = 0.75$ bernilai 46.50%, dari yang awalnya bernilai 10.39 menurun menjadi 4.831 atau terjadi penurunan sebesar 4.831. Sehingga metode *clipping* dan *filtering* ini dapat dikatakan merupakan metode yang efektif dalam menurunkan nilai PAPR.



Gambar 0.10 Grafik Penurunan PAPR Akibat Pengaruh CR pada Modulasi 64QAM

Berdasarkan grafik pada Gambar 4.10, dapat diketahui bahwa semakin kecil nilai *clipping ratio*, reduksi PAPR akan semakin besar, di mana pada saat $CR=0.95$, reduksi PAPR yang dihasilkan sebesar 28.90% sedangkan saat $CR=0.75$, reduksi PAPR yang dihasilkan mencapai 46.50%.

4.6 Analisa Bit Error Rate (BER)

Bit Error Rate (BER) adalah jumlah bit yang diterima dari suatu aliran data melalui jalur komunikasi yang telah berubah karena berbagai hal. Dalam simulasi, BER akan didapatkan setelah sinyal dilewatkan pada kanal AWGN dan diberikan distorsi non linier model Saleh. Sedangkan pada pengukuran dengan WARP, distorsi akan berasal dari komponen-komponen WARP sebagai akibat perubahan gain RF *amplifier*. Gain RF *amplifier* yang terlalu tinggi akan menyebabkan *Power Amplifier* (PA) bekerja di daerah non linier dan salah satu akibat yang ditimbulkan yaitu nilai bit error pada sinyal akan terpengaruh sehingga nilai BER akan naik.

Dalam sub bab ini akan dijelaskan mengenai analisa BER sistem OFDM tanpa reduksi, metode reduksi *clipping*, dan metode *clipping filtering* pada modulasi 16QAM maupun 64QAM.

4.6.1 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping

Sinyal yang dipotong menggunakan teknik *clipping* selanjutnya akan dihitung BERnya dan akan dibandingkan dengan BER sistem OFDM tanpa reduksi. Dari hasil yang didapat dapat diketahui bagaimana pengaruh pemotongan amplitudo sinyal terhadap *Bit Error Rate*. BER sistem OFDM dan BER teknik *clipping* akan dibandingkan dari simulasi dan pengukuran pada WARP, menggunakan modulasi 16QAM dan 64QAM.

4.6.1.1 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping pada Simulasi

Pada simulasi, sinyal yang dikirimkan akan dilewatkan pada kanal AWGN dan akan diberikan *noise* supaya dapat diketahui tanggapan dari sinyal yang dikirimkan apakah mengalami gangguan atau tidak. Selain itu, pada simulasi juga diberikan distorsi non linier model Saleh supaya dapat dianalisa performansi sistem OFDM tanpa reduksi dan dengan reduksi jika diberikan distorsi non linier.

Tabel 4.6 menunjukkan nilai BER sistem OFDM dan teknik *clipping* sedangkan gambar 4.15 menjelaskan tentang perbandingan BER pada sistem OFDM dengan sistem OFDM jika diberikan metode *clipping*. Gambar (a) menggunakan modulasi 16QAM dan Gambar (b) menggunakan modulasi 64QAM.

Seperti yang telah dijelaskan sebelumnya bahwa memotong sinyal dengan *clipping ratio* tertentu akan menyebabkan sinyal menjadi cacat sehingga nilai BER akan naik. Nilai BER yang dihasilkan oleh metode *clipping* dengan modulasi 16QAM dan 64QAM lebih tinggi jika dibandingkan dengan nilai BER OFDM tanpa reduksi. Meskipun demikian, performansi sistem ini masih cukup handal karena perbedaan nilainya tidak terlalu signifikan.

Perlu diketahui bahwa simulasi menggunakan teknik *clipping* ini menggunakan *clipping ratio* sebesar 0.95 karena seperti yang telah dijelaskan di bagian 4.5.1 bahwa semakin rendah nilai *clipping ratio*, maka nilai BER akan semakin tinggi. Untuk meminimalisir nilai BER, dipilih *clipping ratio* 0.95 sehingga nilai BER tidak terlalu tinggi dan dapat dibandingkan dengan BER sistem OFDM tanpa metode reduksi *clipping* dan *filtering*.

Tabel 0.6 Nilai BER Sistem OFDM dan Teknik Clipping pada Simulasi

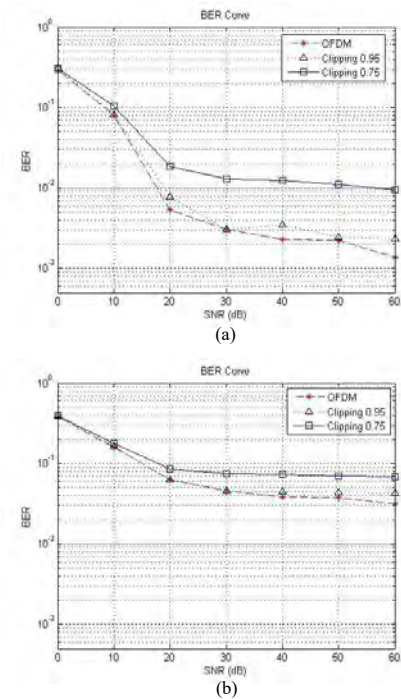
SNR	16QAM			64QAM		
	OFDM	Clipping 0.95	Clipping 0.75	OFDM	Clipping 0.95	Clipping 0.75
0	0.29214	0.308090	0.3057060	0.38024	0.3978742	0.395316
10	0.08011	0.082083	0.1059201	0.16137	0.1635493	0.178773
20	0.00528	0.007847	0.0183333	0.06302	0.0633757	0.086168
30	0.00303	0.003043	0.0130092	0.04591	0.0455941	0.075451
40	0.00231	0.003472	0.0124537	0.03829	0.0453588	0.073753
50	0.00224	0.002442	0.0110185	0.03772	0.0442978	0.070258
60	0.00138	0.002326	0.0095717	0.03153	0.0428626	0.068788

4.6.1.2 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping pada WARP

Pengukuran pada WARP menggunakan kanal *real* sehingga tidak memerlukan kanal AWGN dan distorsi non linier nya pun berasal dari komponen-komponen yang terdapat pada WARP. Tabel 4.7

menunjukkan nilai BER sistem OFDM dan teknik *clipping* sedangkan gambar 4.16 menjelaskan tentang perbandingan BER pada sistem OFDM dan sistem OFDM jika diberikan metode *clipping*. Gambar (a) menggunakan modulasi 16QAM sedangkan Gambar (b) menggunakan modulasi 64QAM.

Hasil yang didapat pada pengukuran dengan WARP relatif sama dengan hasil simulasi. BER yang dihasilkan dari metode *clipping* nilainya lebih besar daripada BER sistem OFDM. Pada modulasi 16QAM, BER OFDM dan *clipping* sama-sama bernilai 0 ketika gain pemancar bernilai 50 hingga 54 dB. Pada gain 56 dB, BER keduanya akan naik namun kenaikan BER *clipping* lebih tinggi daripada BER pada sistem OFDM. Sedangkan pada modulasi 64QAM, walaupun nilai BER metode *clipping* lebih tinggi, namun perbedaannya tidak terlalu signifikan.



Gambar 0.11 Perbandingan BER Simulasi Sistem OFDM dan Metode *Clipping* pada (a) Modulasi 16QAM (b) Modulasi 64QAM

Tabel 0.7 Nilai BER Sistem OFDM dan Teknik Clipping Pengukuran pada WARP

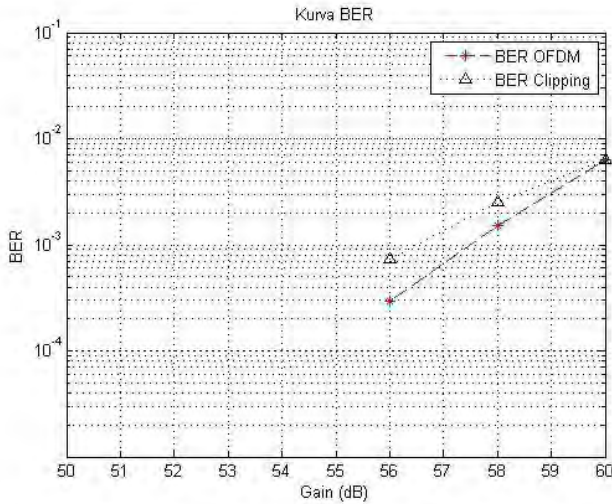
Gain	16QAM		64QAM	
	OFDM	Clipping	OFDM	Clipping
50	0	0	0.000105	0.000154321
52	0	0	0.000573	0.000574846
54	0	0	0.00341	0.003892747
56	0.0002919	0.000734954	0.016551	0.016666667
58	0.0015153	0.002523148	0.043217	0.039915123
60	0.006262	0.006388889	0.086645	0.065933642

4.6.2 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping Filtering

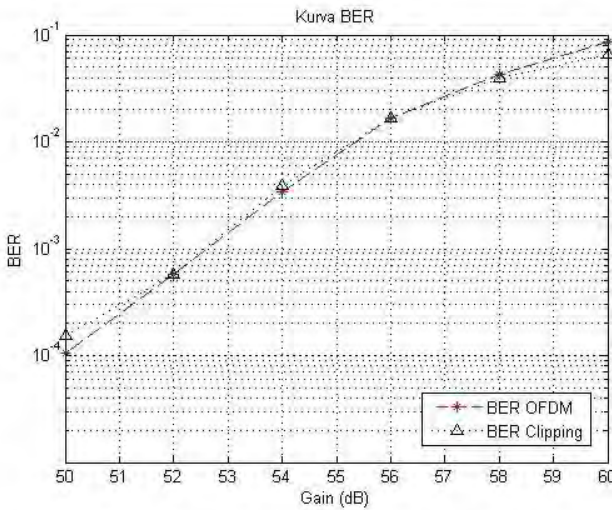
Selain menyebabkan nilai BER naik, teknik clipping juga menyebabkan radiasi *out-of-band* ke kanal-kanal yang berdekatan. Filtering berfungsi untuk mengurangi radiasi *out-of-band* hasil *clipping* sehingga diharapkan nilai BER menjadi lebih baik dibandingkan sebelum diberi teknik *filtering*. Untuk mengetahui hal tersebut, BER sistem OFDM dan BER teknik *clipping filtering* akan dibandingkan dari simulasi dan pengukuran pada WARP, menggunakan modulasi 16QAM dan 64QAM.

Pada proses filtering, sinyal informasi akan dikembalikan ke bentuk semula. Namun, jika terjadi pemotongan yang terlalu banyak pada proses *clipping*, filtering akan sulit mengembalikan sinyal informasi yang dipotong sebelumnya. Sehingga, perlu ada batasan seberapa banyak sinyal dipotong supaya filtering dapat mengurangi nilai BER yang dihasilkan.

Selanjutnya akan dibahas mengenai bagaimana pengaruh penambahan proses *filtering* setelah dilakukan proses *clipping* pada simulasi dan pengukuran pada WARP. Parameter-parameter yang digunakan sama dengan parameter yang digunakan pada proses sebelumnya, baik dalam simulasi maupun pengukuran pada hardware WARP.



(a)



(b)

Gambar 0.12 Perbandingan BER Pengukuran Sistem OFDM dan Metode *Clipping* pada (a) Modulasi 16QAM (b) Modulasi 64QAM

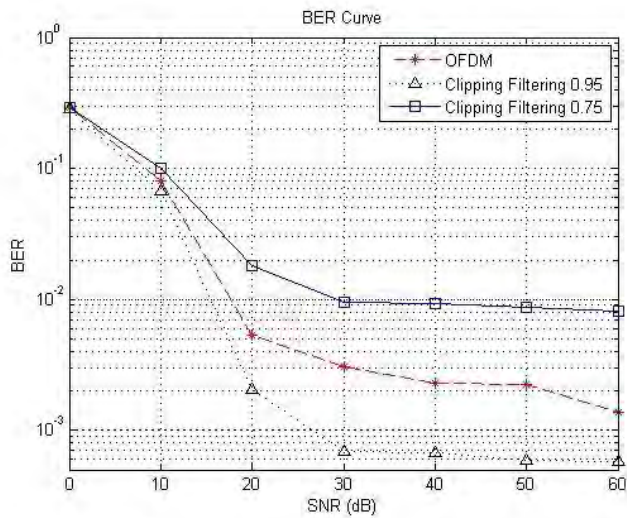
4.6.2.1 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping Filtering pada Simulasi

Pada simulasi ini, penggunaan teknik *filtering* setelah proses *clipping* akan diamati pengaruhnya. Parameter yang diubah-ubah adalah nilai SNR dari 0 hingga 60 dB. Setiap selisih 10 dB akan diamati nilai BER yang dihasilkan. Tabel 4.8 menunjukkan nilai BER sistem OFDM dan teknik *clipping filtering* sedangkan gambar 4.13 menjelaskan tentang perbandingan BER pada sistem OFDM dengan sistem OFDM jika diberikan metode *clipping filtering*. Gambar (a) menggunakan modulasi 16QAM dan Gambar (b) menggunakan modulasi 64QAM.

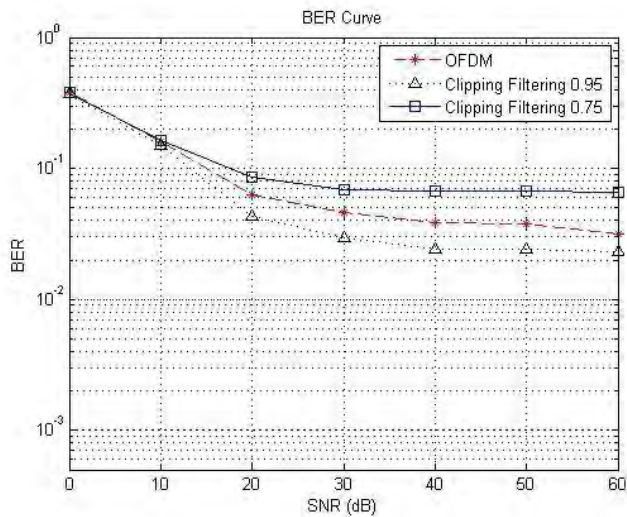
Tabel 0.8 Nilai BER Sistem OFDM dan Teknik *Clipping Filtering* pada Simulasi

SNR	16QAM			64QAM		
	OFDM	Clipping Filtering 0.95	Clipping Filtering 0.75	OFDM	CF 0.95	CF 0.75
0	0.29214	0.283194	0.293663	0.38024	0.36855	0.3772260
10	0.08011	0.067690	0.100121	0.16137	0.15117	0.1643788
20	0.00528	0.002054	0.017922	0.06302	0.04334	0.0860300
30	0.00303	0.000682	0.009490	0.04591	0.02923	0.06875
40	0.00231	0.000671	0.009293	0.03829	0.02398	0.0674344
50	0.00224	0.000590	0.008599	0.03772	0.02392	0.0671219
60	0.00138	0.000578	0.008200	0.03153	0.02326	0.0652044

Pada tabel dapat diketahui bahwa proses *clipping filtering* dapat menurunkan BER yang dihasilkan jika dibandingkan dengan BER yang dihasilkan pada OFDM tanpa metode reduksi. Namun, pada CR=0.75, nilai BER yang dihasilkan lebih tinggi dibandingkan BER pada OFDM. Hal tersebut menunjukkan bahwa *filtering* memiliki batasan untuk dapat mengembalikan sinyal setelah *diclipping*.



(a)



(b)

Gambar 0.13 Perbandingan BER Simulasi Sistem OFDM dan Metode *Clipping Filtering* pada (a) Modulasi 16QAM (b) Modulasi 64QAM

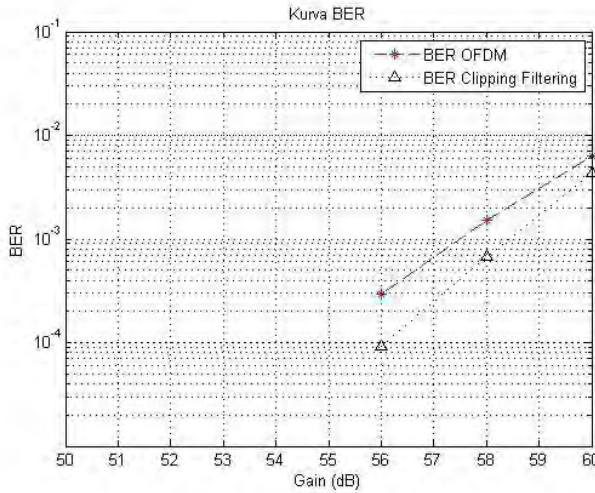
Dari tabel 4.8 didapatkan hasil bahwa pada simulasi perbandingan nilai BER pada sistem OFDM dan reduksi *clipping filtering*, hasilnya BER *clipping filtering* lebih baik. Hal tersebut membuktikan bahwa penambahan teknik *filtering* setelah proses *clipping* dapat memperbaiki BER pada sistem OFDM. Semakin tinggi SNR, nilai BER pada teknik *clipping filtering* akan menunjukkan perubahan BER yang lebih besar dari sebelumnya.

4.6.2.2 Analisa Perbandingan BER Sistem OFDM dan Teknik Clipping Filtering pada WARP

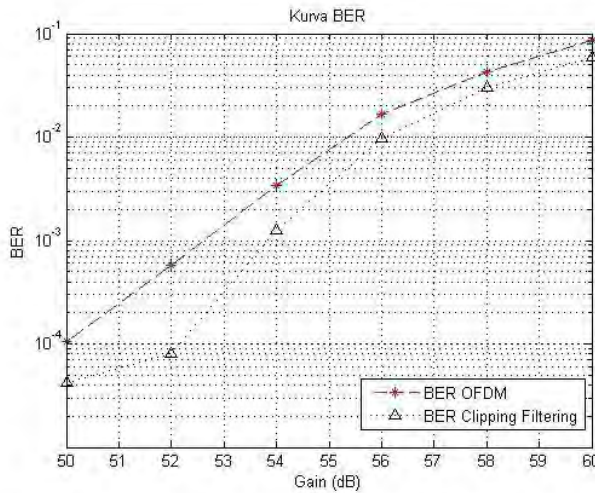
Untuk mengetahui pengaruh teknik *filtering*, pengukuran pada WARP diberikan distorsi berupa perubahan gain pemancar untuk mengetahui berapa nilai BER yang dihasilkan. Gain pemancar akan diubah-ubah dari 50 dB hingga 60 dB dan setiap 2 dB perubahan akan diamati nilai BER nya. Tabel 4.9 menunjukkan nilai BER yang dihasilkan oleh sistem OFDM tanpa reduksi dan sistem OFDM dengan menggunakan teknik *clipping filtering* pada modulasi 16QAM dan 64QAM. Gambar 4.18 menampilkan grafik perbandingan nilai BER pada sistem OFDM dengan sistem OFDM jika diberikan metode *clipping filtering*. Gambar (a) menggunakan modulasi 16QAM dan Gambar (b) menggunakan modulasi 64QAM.

Tabel 0.9 Nilai BER Sistem OFDM dan Teknik *Clipping Filtering* pada Pengukuran

Gain	16QAM		64QAM	
	OFDM	Clipping Filtering	OFDM	Clipping Filtering
50	0	0	0.000105	4.24383E-05
52	0	0	0.000573	8.10185E-05
54	0	0	0.00341	0.001257716
56	0.0002919	9.25926E-05	0.016551	0.009691358
58	0.0015153	0.000677083	0.043217	0.03023534
60	0.006262	0.004375	0.086645	0.058834877



(a)



(b)

Gambar 0.14 Perbandingan BER Pengukuran Sistem OFDM dan Metode *Clipping Filtering* pada (a) Modulasi 16QAM (b) Modulasi 64QAM

Dari tabel 4.9 diketahui bahwa nilai BER metode *clipping filtering* lebih baik daripada BER sistem OFDM. Hal ini menunjukkan bahwa perbaikan kinerja BER akibat pengaruh teknik *filtering* tahan terhadap distorsi non linier yang terjadi pada komponen-komponen WARP akibat perubahan gain pemancar. Pada modulasi 16QAM, BER sama-sama menunjukkan angka 0 pada sistem OFDM dan teknik *clipping filtering* di gain 50 sampai dengan gain 54. Pada saat gain 56 dB, didapatkan perbedaan nilai BER, di mana nilai BER teknik *clipping filtering* menunjukkan hasil yang lebih baik daripada sistem OFDM. Pada modulasi 64QAM, nilai BER yang dihasilkan oleh teknik *clipping filtering* juga lebih baik dibandingkan sistem OFDM tanpa reduksi.

Gambar 4.18 menunjukkan grafik perbandingan BER antara sistem OFDM dan teknik *clipping filtering*. Nilai BER *clipping filtering* menunjukkan penurunan nilai BER namun semakin tinggi nilai gain pemancar, nilainya tidak terlalu signifikan dengan nilai BER yang dihasilkan oleh sistem OFDM tanpa reduksi.

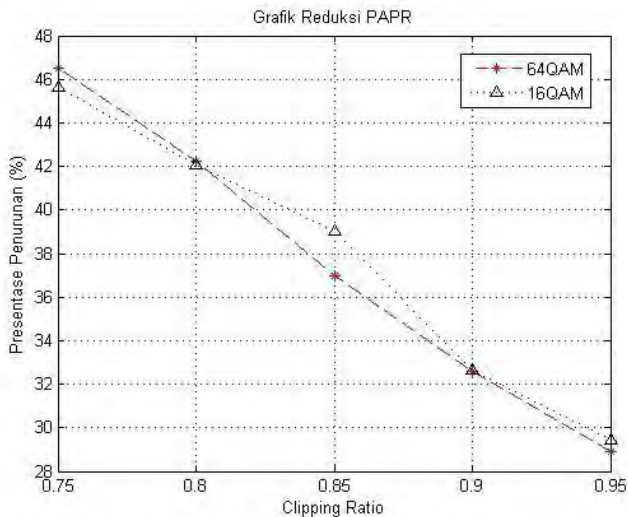
4.7 Analisa Peak to Average Power Ratio (PAPR)

PAPR dihasilkan setelah proses IFFT, yaitu hasil superposisi dari dua atau lebih *subcarrier* dapat menghasilkan variasi daya dengan nilai *peak* yang besar. Hal ini disebabkan oleh modulasi masing-masing *subcarrier* dengan frekuensi yang berbeda-beda. Sehingga apabila beberapa *subcarrier* mempunyai fasa yang koheren, maka akan muncul amplitudo dengan level yang jauh lebih besar dari daya rata-rata sinyalnya, yang selanjutnya disebut dengan *Peak to Average Power Ratio* (PAPR).

Nilai PAPR dihitung setelah melewati proses *clipping*, atau sebelum dikirimkan oleh *transmitter*, sehingga perhitungan nilai PAPR cukup dapat diamati dari simulasi. Tabel 4.10 menampilkan reduksi PAPR yang dihasilkan oleh sistem OFDM dan teknik *clipping filtering* dengan parameter *clipping ratio*. Gambar 4.19 menunjukkan perbandingan penurunan nilai PAPR pada modulasi 16QAM dan 64QAM.

Tabel 0.10 Nilai PAPR Sistem OFDM dan Teknik *Clipping Filtering*

CR	16QAM		64QAM	
	OFDM	Clipping Filtering	OFDM	Clipping Filtering
0.95	10.339	7.296	10.375	7.376
0.90	10.246	6.902	10.338	6.968
0.85	10.643	6.469	10.377	6.534
0.80	10.55	6.111	10.622	6.134
0.75	10.28	5.587	10.39	5.559



Gambar 0.15 Perbandingan Nilai PAPR Sistem OFDM dan Metode *Clipping Filtering* pada Modulasi 16QAM dan 64QAM

Dari grafik tersebut di atas menunjukkan bahwa reduksi PAPR menggunakan modulasi 16QAM lebih besar jika dibandingkan dengan reduksi PAPR pada modulasi 64QAM namun perbedaannya tidak terlalu signifikan.

LAMPIRAN

A. Lembar Pengesahan Proposal

Jurusan Teknik Elektro
Fakultas Teknologi Industri - ITS

TE 141599 TUGAS AKHIR – 4 SKS

Nama Mahasiswa : Fahimah Rahmadian
Nomer Pokok : 2212 100 042
Bidang Studi : Teknik Telekomunikasi Multimedia
Tugas Diberikan : Semester Genap Th. 2015/2016
Dosen Pembimbing : 1. Dr. Ir. Titiek Suryani, MT.
2. Dr. Ir. Suwadi, MT.

10 FEB 2016

Judul Tugas Akhir : **Implementasi dan Analisa Teknik Reduksi PAPR OFDM Menggunakan Metode Clipping dan Filtering pada WARP**
(*Implementation and Analysis PAPR Reduction Technique on OFDM Using Clipping and Filtering Method in WARP*)

Uraian Tugas Akhir :

Proses transmisi sinyal dari pengirim menuju penerima dalam sistem telekomunikasi memiliki gangguan *noise* dan *multipath fading* yang disebabkan oleh pemantulan, *scattering*, atau difraksi. Untuk mengatasinya, salah satunya dengan menggunakan teknik modulasi *multicarrier* dengan tujuan agar sinyal mengalami *flat fading* pada tiap *subcarrier*. Teknik *multicarrier* yang populer digunakan adalah *Orthogonal Frequency Division Multiplexing* (OFDM). Namun, salah satu kekurangan teknik ini adalah nilai *Peak to Average Power Ratio* (PAPR) yang tinggi. PAPR merupakan perbandingan daya puncak dengan daya rata-rata sinyal. Nilai PAPR yang tinggi menyebabkan penurunan performansi sistem OFDM. Oleh karena itu, untuk mengatasinya dibutuhkan suatu metode yang dapat mereduksi nilai PAPR, salah satunya dengan metode *clipping* dan *filtering*.

Pada tugas akhir ini akan diimplementasikan dan dianalisa teknik reduksi PAPR sinyal OFDM menggunakan metode *clipping* dan *filtering* pada WARP yang merupakan salah satu jenis *Software Defined Radio* (SDR).

Dosen Pembimbing I,

Dr. Ir. Titiek Suryani, MT.
NIP. 196411301989032001

Dosen Pembimbing II,

Dr. Ir. Suwadi, MT.
NIP. 196808181993031002

Mengetahui,
Jurusan Teknik Elektro FTI-ITS
Ketua,

Dr. Eng. Ardono Privadi, ST., M.Eng.
NIP. 197309271998031004

Menyetujui,
Bidang Studi Telekomunikasi Multimedia
Koordinator,

Dr. Ir. Endropono, DEA.
NIP. 196504041991021001

B. Listing Program Tugas Akhir

```
% Parameter
USE_WARPLAB_TXRX          = 1;
% Enable WARPLab-in-the-loop (otherwise sim-
only)
SNR_SIMULATION            = 100;
WRITE_PNG_FILES           = 0;
% Enable writing plots to PNG
CHANNEL                   = 11;
% Channel to tune Tx and Rx radios
NODE_TX                   = 2;
NODE_RX                   = 1;
% Waveform params
N_OFDM_SYMS               = 90;
% Number of OFDM symbols
MOD_ORDER                 = 64;
% Modulation order (2/4/16 = BSPK/QPSK/16-QAM)
TX_SCALE                  = 1.0;
% Scale for Tx waveform ([0:1])
INTERP_RATE               = 2;
% Interpolation rate (1 or 2)

% OFDM params
N_SC                      = 64;
% Number of subcarriers
CP_LEN                    = N_SC/4;
% Cyclic prefix length
p1                         = 8;
p2                         = 22;
p3                         = 44;
p4                         = 58;
null                       = [1 28:38];
SC_IND_PILOTS             = [p1    p2    p3    p4];
% Pilot subcarrier indices
SC_IND_DATA               = [2:7  9:21  23:27  39:43
45:57 59:64];
% Data subcarrier indices
N_DATA_SYMS               =
N_OFDM_SYMS*length(SC_IND_DATA);
```

```

% Number of data symbols (one per data-bearing
subcarrier per OFDM symbol)

% Rx processing params
FFT_OFFSET = N_SC/16;
% Number of CP samples to use in FFT (on
average)
LTS_CORR_THRESH = 0.6;
% Normalized threshold for LTS correlation
DO_APPLY_CFO_CORRECTION = 1;
% Enable CFO estimation/correction
DO_APPLY_PHASE_ERR_CORRECTION = 1;
% Enable Residual CFO estimation/correction
DO_APPLY_SFO_CORRECTION = 1;
% Enable SFO estimation/correction

DECIMATE_RATE = INTERP_RATE;

% WARPLab experiment params
USE_AGC = 1;
% Use the AGC if running on WARP hardware
MAX_TX_LEN = 2^19;
% Maximum number of samples to use for this
experiment
SAMP_PADDING = 100;
% Extra samples to receive to ensure both start
and end of waveform visible

if(USE_WARPLAB_TXRX)
    USE_AGC = 0;
    NUMNODES = 2;
    %Create a vector of node objects
    nodes = wl_initNodes(2);
    %Create a UDP broadcast trigger and tell
each node to be ready for it
    eth_trig = wl_trigger_eth_udp_broadcast;

```



```

wl_triggerManagerCmd(nodes,'add_ethernet_trigger',
    '[eth_trig]');
    %Get IDs for the interfaces on the boards.
    Since this example assumes each
    %board has the same interface capabilities,
    we only need to get the IDs
    %from one of the boards
    [RFA,RFB]
wl_getInterfaceIDs(nodes(NODE_TX));

    %Set up the interface for the experiment
wl_interfaceCmd(nodes,'RF_ALL','tx_gains',3,input(
    'GainRF = '));

wl_interfaceCmd(nodes,'RF_ALL','channel',2.4,CHANNEL);

    if(USE_AGC)

wl_interfaceCmd(nodes,'RF_ALL','rx_gain_mode','automatic');
        wl_basebandCmd(nodes,'agc_target',-10);
        wl_basebandCmd(nodes,'agc_trig_delay',
511);
    else

wl_interfaceCmd(nodes,'RF_ALL','rx_gain_mode','manual');
        RxGainRF = 2; %Rx RF Gain in [1:3]
        RxGainBB = 2; %Rx Baseband Gain in
[0:31]

wl_interfaceCmd(nodes,'RF_ALL','rx_gains',RxGainRF,RxGainBB);
    end

        maximum_buffer_len
nodes(NODE_TX).baseband.txIQLen;

```

```

        SAMP_FREQ =
wl_basebandCmd(nodes(NODE_TX), 'tx_buff_clk_freq'
);
    node_tx = nodes(NODE_TX);
    node_rx = nodes(NODE_RX);
    RF_TX = RFA;
    RF_RX = RFA;

    %Set up the baseband for the experiment
    wl_basebandCmd(nodes, 'tx_delay', 0);

wl_basebandCmd(nodes, 'tx_length', maximum_buffer_
len);
    example_mode_string = 'hw';
else
    % Use sane defaults for hardware-dependent
params in sim-only version
    maximum_buffer_len = min(MAX_TX_LEN, 2^20);
    SAMP_FREQ = 40e6;
    example_mode_string = 'sim';
end

%% Define a half-band 2x interpolation filter
response
interp_filt2 = zeros(1,43);
interp_filt2([1 3 5 7 9 11 13 15 17 19 21]) =
[12 -32 72 -140 252 -422 682 -1086 1778 -3284
10364];
interp_filt2([23 25 27 29 31 33 35 37 39 41 43])
= interp_filt2(fliplr([1 3 5 7 9 11 13 15 17 19
21]));
interp_filt2(22) = 16384;
interp_filt2 =
interp_filt2./max(abs(interp_filt2));

% Define the preamble
sts_f = zeros(1,64);
sts_f(1:27) = [0 0 0 0 -1-1i 0 0 0 -1-1i 0 0 0
1+1i 0 0 0 1+1i 0 0 0 1+1i 0 0 0 1+1i 0 0];

```

```

sts_f(39:64) = [0 0 1+1i 0 0 0 -1-1i 0 0 0 1+1i
0 0 0 -1-1i 0 0 0 -1-1i 0 0 0 1+1i 0 0 0];
sts_t = ifft(sqrt(13/6).*sts_f, 64);
sts_t = sts_t(1:16);
% LTS for CFO and channel estimation
lts_f = [0 1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1
1 1 -1 -1 1 -1 1 -1 1 1 1 1 0 0 0 0 0 0 0 0
0 1 1 -1 -1 1 1 -1 1 -1 1 1 1 1 1 -1 -1 1 1 -1
1 -1 1 1 1 1];
lts_t = ifft(lts_f, 64);

% Use 30 copies of the 16-sample STS for extra
AGC settling margin
preamble = [repmat(sts_t, 1, 30) lts_t(33:64)
lts_t lts_t];

% Sanity check inputs
if(SAMP_PADDING + INTERP_RATE*((N_OFDM_SYMS *
(N_SC + CP_LEN)) + length(preamble) + 100) >
maximum_buffer_len)
    fprintf('Too many OFDM symbols for
TX_NUM_SAMPS!\n');
    fprintf('Raise TX_NUM_SAMPS to %d, or \n',
SAMP_PADDING + INTERP_RATE*((N_OFDM_SYMS * (N_SC
+ CP_LEN)) + length(preamble) + 100));
    fprintf('Reduce N_OFDM_SYMS to %d\n',
floor((
(maximum_buffer_len/INTERP_RATE)-
length(preamble) )/( N_SC + CP_LEN )) - 1);
    return;
end

%% Generate a payload
tx_data = randi(MOD_ORDER, 1, N_DATA_SYMS) - 1;

% Functions for data -> complex symbol mapping
(avoids comm toolbox requirement for gammod)
modvec_bpsk = (1/sqrt(2)) .* [-1 1];
modvec_16qam = (1/sqrt(10)) .* [-3 -1 +3 +1];

```

```

modvec_64qam = (1/sqrt(43)) .* [-7 -5 -1 -3 +7
+5 +1 +3];

mod_fcn_bpsk = @ (x)
complex(modvec_bpsk(1+x),0);
mod_fcn_qpsk = @ (x)
complex(modvec_bpsk(1+bitshift(x,
modvec_bpsk(1+mod(x, 2))),
-1)),
mod_fcn_16qam = @ (x)
complex(modvec_16qam(1+bitshift(x,
modvec_16qam(1+mod(x,4))),
-2)),
mod_fcn_64qam = @ (x)
complex(modvec_64qam(1+bitshift(x,
modvec_64qam(1+mod(x,8))),
-3)),

% Map the data values on to complex symbols
switch MOD_ORDER
case 2 % BPSK
tx_syms = arrayfun(mod_fcn_bpsk,
tx_data);
case 4 % QPSK
tx_syms = arrayfun(mod_fcn_qpsk,
tx_data);
case 16 % 16-QAM
tx_syms = arrayfun(mod_fcn_16qam,
tx_data);
case 64 % 64-QAM
tx_syms = arrayfun(mod_fcn_64qam,
tx_data);
otherwise
fprintf('Invalid MOD_ORDER (%d)! Must
be in [2, 4, 16]\n', MOD_ORDER);
return;
end

% Reshape the symbol vector to a matrix with one
column per OFDM symbol
tx_syms_mat = reshape(tx_syms,
length(SC_IND_DATA), N_OFDM_SYMS);

```

```

% Define the pilot tones

% Define the pilot tones
i=sqrt(-1);
switch MOD_ORDER
    case 2          % BPSK
        pilots = ([1 1 1 1].');
    case 4          % QPSK
        pilots = ([1+i      1+i      1+i
1+i].')*(1/sqrt(2));
    case 16         % 16-QAM
        pilots = ([3+i*3      3+i*3      3+i*3
3+i*3].')*(1/sqrt(10));
    case 64         % 64-QAM
        pilots = ([7+i*7      7+i*7      7+i*7
7+i*7].')*(1/sqrt(43));
    otherwise
        fprintf('Invalid MOD_ORDER (%d)! Must
be in [2, 4, 16, 64]\n', MOD_ORDER);
        return;
end% Repeat the pilots across all OFDM symbols
% Repeat the pilots across all OFDM symbols
pilots_mat = repmat(pilots, 1, N_OFDM_SYMS);

%% IFFT

% Construct the IFFT input matrix
ifft_in_mat = zeros(N_SC, N_OFDM_SYMS);

% Insert the data and pilot values; other
subcarriers will remain at 0
ifft_in_mat(SC_IND_DATA, :) = tx_syms_mat;
ifft_in_mat(SC_IND_PILOTS, :) = pilots_mat;
%ofdm_symbol=ifft_in_mat;
ofdm_symbol=ifft_in_mat;
time_domain_signal=(ifft(ofdm_symbol,64));
meano=mean(abs(time_domain_signal).^2);
peako=max(abs(time_domain_signal).^2);

```

```

papro=10*log10(peako./meano);
x=reshape(time_domain_signal,1,[]);
z=x;
PersenA=0.95;%Nilai PersenA bisa diubah

A=PersenA*mean(max(abs(time_domain_signal)));
y=zeros(size(x));
for k=1:length(x)
    if abs(x(k)) <=A
        y(k)=x(k);
    else
        y(k)=A*exp(i*angle(x(k)));
    end
end
signal_Tx=reshape(y,N_SC,[]);

meanz=mean(abs(signal_Tx).^2);
peakz=max(abs(signal_Tx).^2);
paprz=10*log10(peakz./meanz);

%Perform the IFFT
tx_payload_mat = signal_Tx;
signal_tx=fft(signal_Tx,N_SC);
signal_tx(null,:)=0;
ff=0:63;
plot(ff,signal_tx)
pilots_matCLIP=signal_tx(SC_IND_PILOTS, :);
tx_payload_mat=ifft(signal_tx,N_SC); %matikan
ini jika mengamati clipping tanpa filtering
% Insert the cyclic prefix
if(CP_LEN > 0)
    tx_cp = tx_payload_mat((end-CP_LEN+1 : end),
:);
    tx_payload_mat = [tx_cp; tx_payload_mat];
end
% Reshape to a vector
%tx_payload_vec = reshape(tx_payload_mat, 1,[]);
tx_payload_vec= reshape(tx_payload_mat, 1,[]);

```

```

% Construct the full time-domain OFDM waveform
tx_vec = [preamble tx_payload_vec];

% Pad with zeros for transmission
tx_vec_padded = [tx_vec, zeros(1,50)];

%% Interpolate
if(INTERP_RATE == 1)
    tx_vec_air = tx_vec_padded;
elseif(INTERP_RATE == 2)
    tx_vec_2x = zeros(1,
2*numel(tx_vec_padded));
    tx_vec_2x(1:2:end) = tx_vec_padded;
    tx_vec_air = filter(interp_filt2, 1,
tx_vec_2x);
end

% Scale the Tx vector
tx_vec_air2 = TX_SCALE .* tx_vec_air ./
max(abs(tx_vec_air));

%Model Gangguan Nonlinier
hNonlin = comm.MemorylessNonlinearity('Method',
'Saleh model');
tx_vec_air1 = step(hNonlin, tx_vec_air2.);
%tx_vec_air=tx_vec_air1.';
tx_vec_air=tx_vec_air2;
%ratt=mean(tx_payload_vec)/mean(tx_payload_vec2)
;

TX_NUM_SAMPS = length(tx_vec_air);
tic
if(USE_WARPLAB_TXRX)
    wl_basebandCmd(nodes, 'tx_delay', 0);
    wl_basebandCmd(nodes, 'tx_length',
TX_NUM_SAMPS+100);
    % Number of
samples to send

```

```

        wl_basebandCmd(nodes, 'rx_length',
TX_NUM_SAMPS+SAMP_PADDING); % Number of
samples to receive
end

%% WARPLab Tx/Rx
if(USE_WARPLAB_TXRX)
    % Write the Tx waveform to the Tx node
    wl_basebandCmd(node_tx, RF_TX, 'write_IQ',
tx_vec_air(:));

    % Enable the Tx and Rx radios
    wl_interfaceCmd(node_tx, RF_TX, 'tx_en');
    wl_interfaceCmd(node_rx, RF_RX, 'rx_en');

    % Enable the Tx and Rx buffers
    wl_basebandCmd(node_tx, RF_TX,
'tx_buff_en');
    wl_basebandCmd(node_rx, RF_RX,
'rx_buff_en');

    % Trigger the Tx/Rx cycle at both nodes
    eth_trig.send();

    % Retrieve the received waveform from the Rx
node
    rx_vec_air = wl_basebandCmd(node_rx, RF_RX,
'read_IQ');

    rx_vec_air = rx_vec_air(:).';

    %Disable the Tx/Rx radios and buffers

wl_basebandCmd(nodes, 'RF_ALL', 'tx_rx_buff_dis');
    wl_interfaceCmd(nodes, 'RF_ALL', 'tx_rx_dis');
else

```



```

    % Sim-only mode: Apply wireless degradations
    here for sim (noise, fading, etc)

```

```

    % Perfect (ie. Rx=Tx):
    % rx_vec_air = tx_vec_air;

```

```

    % AWGN:
    % rx_vec_air = [tx_vec_air,
zeros(1,SAMP_PADDING)];
    %rx_vec_air = rx_vec_air + 1e-
4*complex(randn(1,length(rx_vec_air)),
randn(1,length(rx_vec_air)));
    % rx_vec_air =
awgn(tx_vec_air,SNR_SIMULATION,'measured');%.*exp
(-pi/8);

```

```

    %Rayleigh
    % ts = 1/40e6;
    % fd = 1;
    % h_ray = rayleighchan(ts,fd);
    % rx_vec_air =
awgn(tx_vec_air,SNR_SIMULATION,'measured').*exp(
-pi/8);
    % rx_vec_air = filter(h_ray,rx_vec_air);
    % CFO:
    % rx_vec_air = tx_vec_air .* exp(-
1i*2*pi*1e-4*[0:length(tx_vec_air)-1]);
end

```

```

toc

```

```

%% Decimate

```

```

if(DECIMATE_RATE == 1)
    raw_rx_dec = rx_vec_air;
elseif(DECIMATE_RATE == 2)
    raw_rx_dec = filter(interp_filt2, 1,
rx_vec_air);
    raw_rx_dec = raw_rx_dec(1:2:end);
end

```

```

%% Correlate for LTS

```

```

% Complex cross correlation of Rx waveform with
time-domain LTS
lts_corr = abs(conv(conj(fliplr(lts_t)),
sign(raw_rx_dec)));

% Skip early and late samples
lts_corr = lts_corr(128:end-128);

% Find all correlation peaks
lts_peaks = find(lts_corr >
LTS_CORR_THRESH*max(lts_corr));

% Select best candidate correlation peak as LTS-
payload boundary
[LTS1, LTS2] = meshgrid(lts_peaks,lts_peaks);
[lts_second_peak_index,y] = find(LTS2-LTS1 ==
length(lts_t));

% Stop if no valid correlation peak was found
if isempty(lts_second_peak_index)
    fprintf('No LTS Correlation Peaks
Found!\n');
    lts_peaks = [566 630];
    lts_second_peak_index = numel(lts_peaks);
    % return;
end

% Set the sample indices of the payload symbols
and preamble
payload_ind =
lts_peaks(max(lts_second_peak_index))+128;
lts_ind = payload_ind-160;

if(DO_APPLY_CFO_CORRECTION)
    %Extract LTS (not yet CFO corrected)
    rx_lts = raw_rx_dec(lts_ind : lts_ind+159);

```

```

    rx_lts1      =      rx_lts(-64+-FFT_OFFSET      +
[97:160]);
    rx_lts2 = rx_lts(-FFT_OFFSET + [97:160]);

    %Calculate coarse CFO est
    rx_cfo_est_lts = mean(unwrap(angle(rx_lts2
.* conj(rx_lts1))));
    rx_cfo_est_lts = rx_cfo_est_lts/(2*pi*64);
else
    rx_cfo_est_lts = 0;
end

% Apply CFO correction to raw Rx waveform
rx_cfo_corr_t      =      exp(-
1i*2*pi*rx_cfo_est_lts*[0:length(raw_rx_dec)-
1]);
rx_dec_cfo_corr = raw_rx_dec .* rx_cfo_corr_t;

% Re-extract LTS for channel estimate
rx_lts = rx_dec_cfo_corr(lts_ind : lts_ind+159);
rx_lts1 = rx_lts(-64+-FFT_OFFSET + [97:160]);
rx_lts2 = rx_lts(-FFT_OFFSET + [97:160]);

rx_lts1_f = fft(rx_lts1);
rx_lts2_f = fft(rx_lts2);

% Calculate channel estimate
rx_H_est = lts_f .* (rx_lts1_f + rx_lts2_f)/2;

%% Rx payload processing

% Extract the payload samples (integral number
of OFDM symbols following preamble)
payload_vec      =      rx_dec_cfo_corr(payload_ind      :
payload_ind+N_OFDM_SYMS*(N_SC+CP_LEN)-1);
payload_mat      =      reshape(payload_vec,
(N_SC+CP_LEN), N_OFDM_SYMS);

```

```

% Remove the cyclic prefix, keeping FFT_OFFSET
samples of CP (on average)
payload_mat_noCP = payload_mat(CP_LEN-
FFT_OFFSET+[1:N_SC], :);

% Take the FFT
syms_f_mat = fft(payload_mat_noCP, N_SC);

% Equalize (zero-forcing, just divide by comped
chan estimates)
syms_eq_mat = syms_f_mat ./ repmat(rx_H_est.',
1, N_OFDM_SYMS);
%RRx_H_est=repmat(conj(rx_H_est).', 1,
N_OFDM_SYMS);
%syms_eq_mat = syms_f_mat .* (RRx_H_est);

if DO_APPLY_SFO_CORRECTION
    % SFO manifests as a frequency-dependent
    phase whose slope increases
    % over time as the Tx and Rx sample streams
    drift apart from one
    % another. To correct for this effect, we
    calculate this phase slope at
    % each OFDM symbol using the pilot tones and
    use this slope to
    % interpolate a phase correction for each
    data-bearing subcarrier.
    pilots_f_mat = syms_eq_mat(SC_IND_PILOTS,
:);
    pilots_f_mat_comp =
pilots_f_mat./pilots_matCLIP;
    pilot_phases =
unwrap(angle(fftshift(pilots_f_mat_comp,1)), [], 1
);
    b = mean(diff(pilot_phases) ./
repmat(mod(diff(fftshift(SC_IND_PILOTS)), 64).', 1
,N_OFDM_SYMS));
    pilot_phase_sfo_corr = fftshift((-
32:31).'*b,1);

```

```

        pilot_phase_corr = exp(-
        1i*(pilot_phase_sfo_corr));

        % Apply the pilot phase correction per
        symbol
        syms_eq_mat = syms_eq_mat .*
        pilot_phase_corr;
    else
        pilot_phase_sfo_corr = zeros(N_SC,
        N_OFDM_SYMS);
    end

    if DO_APPLY_PHASE_ERR_CORRECTION
        % Extract the pilots and calculate per-
        symbol phase error
        pilots_f_mat = syms_eq_mat(SC_IND_PILOTS,
        :);
        pilots_f_mat_comp =
        pilots_f_mat./pilots_matCLIP;
        pilot_phase_err =
        angle(mean(pilots_f_mat_comp));
    else
        pilot_phase_err = zeros(1,N_OFDM_SYMS);
    end
    pilot_phase_err_corr = repmat(pilot_phase_err,
    N_SC, 1);
    pilot_phase_corr = exp(-
    1i*(pilot_phase_err_corr));

    % Apply the pilot phase correction per symbol
    syms_eq_pc_mat = syms_eq_mat .*
    (pilot_phase_corr);
    signal_Rx=syms_eq_pc_mat;
    fft_out=signal_Rx;%fft(signal_Rx,64);
    %fft_out1=fft(signal_Rx1,64);
    data_est1=zeros(48,N_OFDM_SYMS);
    data_est=zeros(48,N_OFDM_SYMS);
    %data_est1=fft_out1(SC_IND_DATA,:);

```

```

symbol_Est=fft_out;
for k=1:N_OFDM_SYMS;
    for m=1:48
        data_est(m,k)=symbol_Est(SC_IND_DATA(m),k);
    end
end
%payload_syms_mat = syms_eq_pc_mat(SC_IND_DATA,
:);

%% Demodulate
%payload_syms_mat=data_est/ratt;
switch MOD_ORDER
    case 2          % BPSK

payload_syms_mat=data_est/(mean(max(abs(data_est
)))));
        case 4          % QPSK

payload_syms_mat=(sqrt(2*1*1))*data_est/(mean(max
(abs(data_est))))/sqrt(2);
        case 16          % 16-QAM

payload_syms_mat=(sqrt(2*3*3))*data_est/(mean(max
(abs(data_est))))/sqrt(10);
        case 64          % 64-QAM

payload_syms_mat=(sqrt(2*7*7))*data_est/(mean(max
(abs(data_est))))/sqrt(43);
        otherwise
            fprintf('Invalid MOD_ORDER (%d)! Must
be in [2, 4, 16]\n', MOD_ORDER);
            return;
end
rx_syms = reshape(payload_syms_mat, 1,
N_DATA_SYMS);

demod_fcn_bpsk = @(x) double(real(x)>0);
demod_fcn_qpsk = @(x) double(2*(real(x)>0) +
1*(imag(x)>0));

```

```

demod_fcn_16qam = @(x) (8*(real(x)>0)) +
(4*(abs(real(x))<0.6325)) + (2*(imag(x)>0)) +
(1*(abs(imag(x))<0.6325));
demod_fcn_64qam = @(x) (32*(real(x)>0)) +
(16*(abs(real(x))<0.6172)) +
(8*((abs(real(x))<(0.9258))&&((abs(real(x))>(0.3
086)))) + (4*(imag(x)>0)) +
(2*(abs(imag(x))<0.6172)) +
(1*((abs(imag(x))<(0.9258))&&((abs(imag(x))>(0.3
086)))));

switch(MOD_ORDER)
    case 2 % BPSK
        rx_data = arrayfun(demod_fcn_bpsk,
rx_syms);
    case 4 % QPSK
        rx_data = arrayfun(demod_fcn_qpsk,
rx_syms);
    case 16 % 16-QAM
        rx_data = arrayfun(demod_fcn_16qam,
rx_syms);
    case 64 % 64-QAM
        rx_data = arrayfun(demod_fcn_64qam,
rx_syms);
end

%% Plot Results
cf = 0;

% Tx signal
cf = cf + 1;
figure(cf); clf;

subplot(2,1,1);
plot(real(z), 'b');
axis([0 length(z) -TX_SCALE TX_SCALE])
grid on;
title('Tx Waveform (I) before Clipping');

```

```

subplot(2,1,2);
plot(imag(z), 'r');
axis([0 length(z) -TX_SCALE TX_SCALE])
grid on;
title('Tx Waveform (Q) before Clipping');

if(WRITE_PNG_FILES)
    print(gcf,sprintf('wl_ofdm_plots_%s_txIQ',
example_mode_string), '-dpng', '-r96', '-
painters'))
end

% Rx signal
cf = cf + 1;
figure(cf); clf;
subplot(2,1,1);
plot(real(tx_payload_vec), 'b');
axis([0 length(tx_payload_vec) -TX_SCALE
TX_SCALE])
grid on;
title('Rx Waveform (I) after Clipping');

subplot(2,1,2);
plot(imag(tx_payload_vec), 'r');
axis([0 length(tx_payload_vec) -TX_SCALE
TX_SCALE])
grid on;
title('Rx Waveform (Q) after Clipping');

if(WRITE_PNG_FILES)
    print(gcf,sprintf('wl_ofdm_plots_%s_rxIQ',
example_mode_string), '-dpng', '-r96', '-
painters'))
end

% Rx LTS correlation
cf = cf + 1;
figure(cf); clf;
lts_to_plot = lts_corr;

```



```

plot(lts_to_plot, '-b', 'LineWidth', 1);
hold on;
grid on;
line([1                                length(lts_to_plot)],
LTS_CORR_THRESH*max(lts_to_plot)*[1                                1],
'LineStyle', '--', 'Color', 'r', 'LineWidth',
2);
title('LTS Correlation and Threshold')
xlabel('Sample Index')
myAxis = axis();
axis([1, 1000, myAxis(3), myAxis(4)])

if(WRITE_PNG_FILES)

print(gcf,sprintf('wl_ofdm_plots_%s_ltsCorr',
example_mode_string), '-dpng', '-r96', '-
painters')
end

% Channel Estimates
cf = cf + 1;

rx_H_est_plot                                     =
repmat(complex(NaN,NaN),1,length(rx_H_est));
rx_H_est_plot(SC_IND_DATA)                       =
rx_H_est(SC_IND_DATA);
rx_H_est_plot(SC_IND_PILOTS)                     =
rx_H_est(SC_IND_PILOTS);

x = (20/N_SC) * (-(N_SC/2):(N_SC/2 - 1));

figure(cf); clf;
subplot(2,1,1);
stairs(x - (20/(2*N_SC)),
fftshift(real(rx_H_est_plot)), 'b', 'LineWidth',
2);
hold on

```

```

stairs(x - (20/(2*N_SC)),
fftshift(imag(rx_H_est_plot)), 'r', 'LineWidth',
2);
hold off
axis([min(x) max(x) -1.1*max(abs(rx_H_est_plot))
1.1*max(abs(rx_H_est_plot))])
grid on;
title('Channel Estimates (I and Q)')

subplot(2,1,2);
bh = bar(x,
fftshift(abs(rx_H_est_plot)),1,'LineWidth', 1);
shading flat
set(bh,'FaceColor',[0 0 1])
axis([min(x) max(x) 0
1.1*max(abs(rx_H_est_plot))])
grid on;
title('Channel Estimates (Magnitude)')
xlabel('Baseband Frequency (MHz)')

if(WRITE_PNG_FILES)

print(gcf,sprintf('wl_ofdm_plots_%s_chanEst',
example_mode_string), '-dpng', '-r96', '-
painters')
end

%% Pilot phase error estimate
cf = cf + 1;
figure(cf); clf;
subplot(2,1,1)
plot(pilot_phase_err, 'b', 'LineWidth', 2);
title('Phase Error Estimates')
xlabel('OFDM Symbol Index')
ylabel('Radians')
axis([1 N_OFDM_SYMS -3.2 3.2])
grid on

h = colorbar;

```

```

set(h, 'Visible', 'off');

subplot(2,1,2)
imagesc(1:N_OFDM_SYMS, (SC_IND_DATA - N_SC/2),
fftshift(pilot_phase_sfo_corr,1))
xlabel('OFDM Symbol Index')
ylabel('Subcarrier Index')
title('Phase Correction for SFO')
colorbar
myAxis = caxis();
if(myAxis(2)-myAxis(1) < (pi))
    caxis([-pi/2 pi/2])
end

if(WRITE_PNG_FILES)

print(gcf, sprintf('wl_ofdm_plots_%s_phaseError',
example_mode_string), '-dpng', '-r96', '-
painters')
end

%% Symbol constellation
cf = cf + 1;
figure(cf); clf;
ax=(1/sqrt(43)) .* [-6 -6;-4 -4;-2 -2;0 0;2 2;4
4;6 6];
bx=[-1.5 1.5;-1.5 1.5;-1.5 1.5;-1.5 1.5;-1.5
1.5;-1.5 1.5;-1.5 1.5]

plot(payload_syms_mat(:), 'ro', 'MarkerSize', 1);
axis square; axis((1/sqrt(43))*[-10 10 -10 10]);
%axis square; axis(1.5*[-1 1 -1 1]);
%grid on;
hold on;
plot(bx', ax', 'k:', ax', bx', 'k:')
hold on
plot(tx_syms_mat(:), 'bo', 'LineWidth', 2);
title('Tx and Rx Constellations')

```

```

h = legend('Rx','Tx','Location','EastOutside');
% tmp = get(h,'Children');
% set(tmp(8),'MarkerSize',6)
% set(tmp(8),'MarkerFaceColor','r')

if(WRITE_PNG_FILES)

print(gcf,sprintf('wl_ofdm_plots_%s_constellatio
ns', example_mode_string), '-dpng', '-r96', '-
painters')
end

% EVM & SNR
cf = cf + 1;
figure(cf); clf;

evm_mat      =      abs(payload_syms_mat      -
tx_syms_mat).^2;
aevms = mean(evm_mat(:));
snr = 10*log10(1./aevms);

subplot(2,1,1)
plot(100*evm_mat(:),'o','MarkerSize',1)
axis tight
hold on
plot([1      length(evm_mat(:))],      100*[aevms,
aevms],'r','LineWidth',4)
myAxis = axis;
h      =      text(round(.05*length(evm_mat(:))),
100*aevms+      .1*(myAxis(4)-myAxis(3)),
sprintf('Effective SNR: %.1f dB', snr));
set(h,'Color',[1 0 0])
set(h,'FontWeight','bold')
set(h,'FontSize',10)
set(h,'EdgeColor',[1 0 0])
set(h,'BackgroundColor',[1 1 1])

```

```

hold off
xlabel('Data Symbol Index')
ylabel('EVM (%)');
h = legend('Per-Symbol EVM', 'Average EVM', 'Location', 'NorthWest');
tmp = get(h, 'Children');

% set(tmp(8), 'MarkerSize', 6)
% set(tmp(8), 'MarkerFaceColor', 'b')

title('EVM vs. Data Symbol Index')
grid on

subplot(2,1,2)
imagesc(1:N_OFDM_SYMS, (SC_IND_DATA - N_SC/2),
100*fftshift(evm_mat,1))

grid on
xlabel('OFDM Symbol Index')
ylabel('Subcarrier Index')
title('EVM vs. (Subcarrier & OFDM Symbol)')
h = colorbar;
set(get(h, 'title'), 'string', 'EVM (%)');
myAxis = caxis();
if (myAxis(2)-myAxis(1)) < 5
    caxis([myAxis(1), myAxis(1)+5])
end

if(WRITE_PNG_FILES)
    print(gcf, sprintf('wl_ofdm_plots_%s_evm',
example_mode_string), '-dpng', '-r96', '-
painters')
end

%[ofdm_symbol(:,1) symbol_Est(:,1)]

```

```

%% Calculate Rx stats

sym_errs = sum(tx_data ~= rx_data);
bit_errs = length(find(dec2bin(bitxor(tx_data,
rx_data),8) == '1'));
rx_evm = sqrt(sum((real(rx_syms) -
real(tx_syms)).^2 + (imag(rx_syms) -
imag(tx_syms)).^2)/(length(SC_IND_DATA) *
N_OFDM_SYMS));

fprintf('\nResults:\n');
fprintf('Num Bytes:      %d\n', N_DATA_SYMS *
log2(MOD_ORDER) / 8);
fprintf('Sym Errors:      %d (of %d total
symbols)\n', sym_errs, N_DATA_SYMS);
fprintf('Bit Errors:      %d (of %d total bits)\n',
bit_errs, N_DATA_SYMS * log2(MOD_ORDER));

cfo_est_lts =
rx_cfo_est_lts*(SAMP_FREQ/INTERP_RATE);
cfo_est_phaseErr =
mean(diff(unwrap(pilot_phase_err)))/(4e-6*2*pi);
cfo_total_ppm = ((cfo_est_lts +
cfo_est_phaseErr) / ((2.412+(.005*(CHANNEL-
1)))*1e9)) * 1e6;

fprintf('CFO Est:      %3.2f kHz (%3.2f ppm)\n',
(cfo_est_lts + cfo_est_phaseErr)*1e-3,
cfo_total_ppm);
fprintf('          LTS      CFO      Est:
%3.2f kHz\n', cfo_est_lts*1e-3);
fprintf('          Phase Error Residual CFO Est:
%3.2f kHz\n', cfo_est_phaseErr*1e-3);

if DO_APPLY_SFO_CORRECTION
    drift_sec = b / (2*pi*312500);
    sfo_est_ppm = 1e6*mean((diff(drift_sec) /
4e-6));
    sfo_est = sfo_est_ppm*20;

```

```

        fprintf('SFO Est:           %3.2f Hz (%3.2f
ppm)\n', sfo_est, sfo_est_ppm);

end

%Grafik ccdf
cf = cf + 1;
figure(cf); clf;
[cy,cx]=ccdf(papro,0.1);
semilogy(cx,cy)
axis([0 12 0.0001 1])
grid on
hold on
[cy,cx]=ccdf(paprz,0.1);
semilogy(cx,cy,'r')
axis([0 12 0.0001 1])
title('Peak to Average Power Ratio')
xlabel('PAPR')
ylabel('CCDF')

if(WRITE_PNG_FILES)
    print(gcf,sprintf('wl_ofdm_plots_%s_ccdf',
example_mode_string), '-dpng', '-r96', '-
painters')
end

```

C. Langkah-langkah Pengoperasian WARP

Berikut adalah langkah-langkah bagaimana mengoperasikan WARP sebelum alat ini digunakan dalam pengukuran:

1. Pasang semua antenna pada radio *daughterboard*



2. Pasang CF pada slot CF di modul WARP



3. Hubungkan kabel UTP pada slot RG45 di modul WARP



4. Hubungkan kabel UTP pada komputer



5. Hubungkan kabel UTP pada *Ethernet switch*



6. Hubungkan kabel *power supply* pada *port supply* di modul WARP



7. Pastikan hubungan antara 2 unit modul WARP dan komputer seperti pada Gambar 3.16

8. Jika semua telah terhubung, hidupkan power modul WARP



BAB 5

PENUTUP

5.1 Kesimpulan

Setelah dilakukan implementasi dan analisa teknik reduksi PAPR OFDM menggunakan teknik *clipping* dan *filtering* pada WARP, maka secara keseluruhan dapat diambil kesimpulan sebagai berikut:

1. Semakin kecil nilai *clipping ratio* maka semakin banyak sinyal yang dipotong, sinyal akan menjadi cacat dan akan menyebabkan nilai BER naik.
2. Semakin kecil nilai *clipping ratio*, maka reduksi PAPR yang dihasilkan semakin besar. Pada saat $A=0.75$, reduksi PAPR bisa mencapai $\geq 40\%$.
3. Nilai BER yang dihasilkan teknik *clipping* lebih tinggi jika dibandingkan dengan sinyal OFDM karena *clipping* dapat menyebabkan distorsi *in-band* yang akan menyebabkan nilai BER akan naik. Sedangkan pada teknik *clipping* dan *filtering*, nilai BER yang dihasilkan lebih rendah karena *filtering* dapat mengurangi radiasi *out-of-band*.
4. Perbandingan kinerja sistem OFDM ditunjukkan pada nilai BER yang diperoleh dengan mengubah SNR pada simulasi. Nilai BER akan semakin kecil terhadap bertambahnya SNR. Dengan kata lain, nilai BER berbanding terbalik terhadap SNR.
5. Nilai BER akan semakin tinggi ketika gain pemancar pada pengukuran WARP dinaikkan. Dengan kata lain, nilai BER berbanding lurus terhadap gain pemancar.

5.2 Saran

Dari tugas akhir yang telah dilakukan, masih terdapat beberapa hal yang dapat disempurnakan. Berikut saran-saran untuk pengembangan sistem reduksi PAPR sinyal OFDM menggunakan metode *clipping filtering*:

1. Simulasi dan implementasi dapat dikembangkan untuk ukuran tipe *clipping* yang berbeda.
2. Simulasi dan implementasi dapat dikembangkan untuk gangguan sinyal yang lebih bervariasi.
3. Simulasi dapat dikembangkan dengan menyertakan parameter – parameter sistem komunikasi digital lainnya.

DAFTAR PUSTAKA

- [1] Cho, Yong Soo, Won Young Yang, Chung-Gu Kang. *MIMO-OFDM Wireless Communicatios with MATLAB*. Singapore. 2010.
- [2] S. Bavi And S. Dhotre, *PAPR Reduction in OFDM System Using Clipping and Filtering Method*, IJARCSSE, Vol. 5, Issue 2, Februari 2015.
- [3] Dhyaksa, IBGD, Wirastuti, NMAED, Widyantara, IMO. *Reduksi Peak to Average Power Ratio (PAPR) pada DFT-OFDM Menggunakan Teknik Clipping Filtering*, E-Journal SPEKTRUM, Vol. 1, No. 1, Desember 2014.
- [4] X.Li And L.J.Cimini, *Effects of Clipping And Filtering On The Performance Of OFDM*, IEEE Communications Letters, Vol. 2, No. 5, pp.131-133, May 1998.
- [5] S. Albdran, A. Alshammari, and M. Matin, *Clipping and Filtering for Reducing PAPR in OFDM*, IOSR Journal of Engineering, Vol. 2, Issue 9, PP 91-97, September 2012.
- [6] Saini, Sanjeev, Dr. O. P. Sahu, *Peak to Average Power Ratio Reduction in OFDM System by Clipping and Filtering*, IJECCCT, Vol. 2, Issue 3, May 2012.
- [7] Prafulla D. Gawande And Sirdmath A. Ladhake, *PAPR Performance of OFDM System Using Clipping and Filtering Method*, IJAET, Vol.6, Issue 2, pp.789-794, May 2013.
- [8] Wang, Luqing, Tellambura, Chinttha, *A Simplified Clipping and Filtering Technique for PAPR Reduction in OFDM System*, IEEE Signal Processing Letters, Vol. 12, No. 6, June 2005.
- [9] Sawada, Manabu. Okada Hiraku. Yamazato, Takaya. Katayama, Masaaki, *Influence of ADC Nonlinearity on the Performance of an OFDM Receiver*, IEICE, Vol. E89-B, No. 12, December 2006.
- [10] M. A. Jarajreh. J. L. Wei. J. M. Tang. Z. Ghassemlooy. W. P. Ng, *Effect of Number of Sub-carriers, Cyclic Prefix, and Analogue to Digital Converter Parameters on Coherent Optical Orthogonal Frequency Division Multiplexing Modem's Transmission Performance*, IETDL, Vol. 4, Issue. 2, pp. 213-222, April 2009.

RIWAYAT HIDUP



Penulis dilahirkan di Jombang pada tanggal 2 Februari 1995. Telah menyelesaikan sekolah dasar di SDN Mojo VI No. 225 Surabaya. Sekolah menengah pertama di SMPN 29 Surabaya sampai tahun 2009, sekolah menengah atas di SMAN 15 Surabaya sampai tahun 2012. Lalu melanjutkan pendidikan di S1 Teknik Elektro Institut Teknologi Sepuluh Nopember pada tahun 2012 program studi Telekomunikasi Multimedia.

Penulis dapat dihubungi melalui alamat e-mail fahimahrahmadian@gmail.com.